

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Digitalna karakterizacija z uporabo računalniškega vida v realnem času

Robert Ravnik

Delo je pripravljeno v skladu
s Pravilnikom o podeljevanju Prešernovih nagrad študentom,

pod mentorstvom
prof. dr. Franca Soline

in somentorstvom
dr. Boruta Batagelja.

Ljubljana, 2009

Posvetilo

00111100	01101001	00111110	00100000
01010010	01100001	01100100	01101111
01110110	01100101	01100100	01101110
01100101	01101101	01110101	00100000

01101001	01101110	00100000	01110101
01110011	01110100	01110110	01100001
01110010	01101010	01100001	01101100
01101110	01100101	01101101	01110101

00100000	01110010	01100001	01100011
01110101	01101110	01100001	01101100
01101110	01101001	01110011	01101011
01100101	01101101	01110101	00100000

01110101	01101101	01110101	00100000
00101110	00101110	00101110	00100000
00111100	00101111	01101001	00111110
00100000	00111010	00101101	00101001

Kazalo

Povzetek	1
1 Uvod	5
2 Digitalno oglaševanje in digitalna karakterizacija	9
2.1 Digitalno oglaševanje	9
2.2 Digitalna karakterizacija	11
3 Sledenje obrazov v realnem času	13
3.1 Detektor obrazov Viola in Jones	13
3.2 Selekcija in validacija obrazov	17
3.3 Sledenje obrazov v realnem času	21
3.4 Ocena razdalje	22
3.5 Implementacija	24
3.5.1 Programska knjižnica OpenCV	24
3.5.2 Programska knjižnica Boost	24
4 Karakterizacija obrazov	27
4.1 Metoda glavnih komponent	28
4.2 Klasifikacija spola	31
4.2.1 Projekcija slik z metodo glavnih komponent	31
4.2.2 Učenje klasifikatorja v okolju Orange	33
5 Zajem in vizualizacija podatkov	43
5.1 Shranjevanje podatkov	44
5.2 Vizualizacija podatkov in generiranje poročil	45
6 Zaključek	49
A Prevod posvetila	51

Seznam slik	53
Seznam tabel	55
Seznam algoritmov	57
Literatura	59

Seznam uporabljenih kratic in simbolov

API - Programski vmesnik (angl. Application Program Interface)

HTML - Označevalni jezik za oblikovanje večpredstavnostnih dokumentov, ki omogoča povezave znotraj dokumenta ali med dokumenti (angl. Hyper Text Markup Language)

HTTP - Protokol za izmenjavo besedil ter grafičnih, zvočnih in drugih večpredstavnostnih vsebin na spletu (angl. Hyper Text Transfer Protocol)

LAN - Krajevno oziroma lokalno omrežje (angl. Local Area Network)

PCA - Metoda glavnih komponent (angl. Principal Component Analysis)

PHP - Skriptni programski jezik, namenjen za izdelavo dinamičnih spletnih strani (angl. Hypertext Pre-Processor)

SVD - Metoda razcepa na singularne vrednosti (angl. Singular Value Decomposition)

XML - Razširljivi označevalni jezik (angl. Extensible Markup Language)

Povzetek

Metode računalniškega vida, povezane z moderno računalniško opremo, omogočajo nove možnosti za posredovanje vizualnih informacij. S tem, da računalniški sistem določi nekatere lastnosti iz slike obraza opazovalca (npr. spol, starost), mu lahko prilagodi prikazano informacijo glede na te lastnosti. Za oznako procesa zaznavanja, določevanja značilnosti gledalca in posredovanja povratne informacije v tem delu vpeljemo izraz "digitalna karakterizacija", saj naš sistem za digitalno karakterizacijo bistveno presega dosedanje sisteme za digitalno oglaševanje (angl. digital signage), ki zgolj posredujejo informacijo opazovalcu, a ga ne zaznavajo, ne določijo njegovih lastnosti in mu ne prilagodijo prikazane informacije. Predstavljen je izviren računalniški sistem za zaznavanje oseb in določevanje njihovih osnovnih značilnosti (npr. spola) na osnovi slike, ki jo posreduje video kamera. Sistem posreduje osebi tudi povratno informacijo preko računalniškega ali informacijskega zaslona, ki je nameščen skupaj s kamero. Povratna informacija je prilagojena značilnostim osebe - gledalca. Osnovni komponenti sistema sta predvajalno mesto in glavni strežnik. Na predvajalnem mestu se s pomočjo metod računalniškega vida izvaja karakterizacija obrazov opazovalcev, na osnovi česar se lahko prilagodi prikazovana vsebina. Glavni strežnik določa vsebine za prikazovanje in skrbi za manipulacijo z zajetimi podatki. Detekcija in sledenje več opazovalcev se izvaja v realnem času. Pri karakterizaciji obraza opazovalca uporabimo metodo glavnih komponent (PCA) in okolje za podatkovno rudarjenje Orange. Posebej obdelamo klasifikacijo spola iz obraza opazovalca. Za ta namen v okviru okolja Orange uporabimo in primerjamo metode strojnega učenja: naivni Bayes, K-najbližjih sosedov, klasifikacijsko drevo in naključni gozdovi. Kot učno in testno množico uporabimo del zbirke obraznih slik FERET. Z metodo naključnih gozdov lahko določimo spol opazovalca s 83,3% zanesljivostjo v 17,9 ms. Za upravljanje sistema razvijemo spletno aplikacijo, ki omogoča dinamično generiranje poročil, statistično obdelavo in vizualizacijo zajetih podatkov. Sistem je kompatibilen z vsemi običajnimi operacijskimi sistemi in zasnovan za izvajanje na standardni računalniški ter video opremi, zato je uporaben povsod, kjer želimo informacijo prilagoditi gledalcu: v izobraževanju, zdravstvu, oglaševanju itd.

Ključne besede:

digitalno oglaševanje, digitalna karakterizacija, računalniški vid, klasifikacija spola.

Abstract

Digital signage in combination with computer vision opens new possibilities for out-of-home advertisement. By determining certain characteristics of observers (e.g. gender, age) the information presented to them may be adjusted according to their expected preferred interest. To denote the process of detecting, characterizing and responding to the observer we introduce the term “digital characterization” as our system significantly reaches over the existing so called “digital signage” systems that only presents information without detecting and characterizing the observers. Digital characterization in real time has become possible due to the development of the image analysis methods and the development of the widely available computer equipment. A computer system for digital characterization is described. It is an intelligent system for displaying selected visual information on a computer screen. The system tracks and characterizes the viewers by analyzing the images of their faces taken by a camera attached to the screen, using the computer vision methods in real time. It also performs logging and analysis of recorded data. Main components of the system are the player and the main server. The player performs characterization of the viewers which can be used for adjusting the information on the screen. The main server performs selection of the presented information and manipulates the data. Detection and tracking of several observers is performed in real time. In face characterization we use the Principal Component Analysis (PCA) and the data mining environment Orange. Characterization of the observer’s gender from their face is specially treated. The following methods of machine learning are applied and inter-compared in the Orange environment: naive Bayes, K-nearest neighbors, classification tree and random forest. A part of FERET face image library is used as learning and test sets. Selected methods are implemented for real time application on a PC. Gender of an observer can be determined in 17,9 ms with 83,3% reliability using random forest classifier. A web application for managing of the system and for generating reports is developed, including statistical analysis and visualization of data. The system is designed for standard computer and video equipment. For this reason it is practically applicable wherever we want to adapt the information to the viewer: in education, medical institutions, marketing, etc.

Key words:

digital signage, digital characterization, computer vision, gender classification.

Poglavje 1

Uvod

Digitalni oziroma računalniško vodeni sistemi postajajo sestavni del našega življenja. Kot primer digitalnega sistema navedimo zaslon za oglaševanje ali predvajanje določene informacije, ki ga upravlja računalnik. Digitalni sistem sestavljajo različne računalniške in video komponente, njegovo bistvo pa je računalniški program, ki izvaja in upravlja proces. Ker so digitalni sistemi namenjeni široki uporabi, morajo biti zasnovani na preprostih in cenениh računalniških komponentah. Zato pa morajo biti računalniški programi toliko bolj učinkoviti in zahtevni. V svetu potekajo intenzivne raziskave in razvoj metod, ki so njihova podlaga.

Osnovi namen tega dela je raziskava in izboljšava metod ter praktična izdelava računalniškega programa, ki upravlja digitalni sistem za prikazovanje vsebine v javnem prostoru, npr. v nakupovalnem centru. Tak sistem je postavljen na vidnem mestu in podaja slikovno ali tekstualno informacijo množici naključnih opazovalcev, ki se spreminja v relativno kratkem času. Za ilustracijo navedimo primer informacijskega zaslona v sprejemni pisarni bolnišnice, ki čakajočim prikazuje preventivne zdravstvene informacije ali pa zaslon, ki vstopajočim v prodajalno reklamira izdelke. Če se taki sistemi uporabljajo za oglaševanje, jih imenujemo sistemi za *digitalno oglaševanje*. V tuji literaturi se je zanje udomačil izraz “digital signage” [11, 15, 25]. Ker so postali LCD zasloni in ostale računalniške komponente poceni in zanesljivi, se sistemi za digitalno oglaševanje že množično uporabljajo v številnih javnih prostorih kot so nakupovalna središča, letališča in celo poštni uradi.

V prvem sklopu dela opredelimo osnovno zgradbo in princip delovanja sistema za digitalno zaznavanje. S pomočjo metod računalniškega vida [26] ga izboljšamo tako, da vgradimo metode za prepoznavanje osnovnih značilnosti opazovalca (npr. spol, starost, razpoloženje) z namenom, da opazovalcu pri-

kažemo zanj prilagojeno vsebino. Inteligentni digitalni sistem, ki vsebuje opisane lastnosti, imenujemo sistem digitalne karakterizacije [22]. Pojem *digitalna karakterizacija* vpeljemo, ker se tak sistem bistveno loči od sistemov za digitalno oglaševanje, saj poleg tega, da zazna (detektira) osebo, določi tudi njene osnovne značilnosti ter ji posreduje povratno, specifično prilagojeno informacijo. Sistem v sprejemni pisarni bolnišnice prikaže drugačne preventivne zdravstvene informacije, če ugotovi, da se pred zaslonom nahaja starejša gospa, kot če ugotovi, da je tam mlad fant. Sistem na vhodu v trgovino pa bo poleg gledalcu prilagojenih oglasov beležil tudi statistične podatke o kupcih (npr. kdaj jih največ nakupuje, kako so porazdeljeni po starostnih skupinah in spolu), na osnovi katerih trgovina lahko prilagodi svoj delovni ritem in ponudbo.

Pri digitalni karakterizaciji je pomembno zagotavljati varovanje osebnih podatkov gledalca v skladu z zakoni in predpisi [28]. Zato poskrbimo, da sistem beleži samo kumulativne statistike, vsi osebni podatki (slika, spol, starost) pa se zbršejo takoj po karakterizaciji.

Za uporabo pri digitalni karakterizaciji raziščemo in uporabimo različne metode računalniškega vida. Računalniški vid je področje računalništva, ki se ukvarja s procesiranjem in analizo slik [26]. Računalnik interpretira sliko na osnovi metod razpoznavanja vzorcev in iskanja značilnic. Obsežen nabor metod nam omogoča zajem specifičnih informacij. V okviru dela se omejimo na metode za iskanje in sledenje obrazov ter obraznih značilnic [14]. Uporaba običajne strojne opreme ter dinamika sistema zahtevata uporabo metod računalniškega vida majhne časovne zahtevnosti. Predstavimo metodo AdaBoost [27], ki omogoča detekcijo objektov v realnem času. Opišemo program za detekcijo in sledenje obrazov, ki smo ga razvili na osnovi teh metod. S pomočjo programa preštajemo število ljudi, ki v nekem obdobju gleda prikazovano vsebino, določimo čas zadrževanja pred zaslonom ter oddaljenost opazovalca od zaslona. Pri konstrukciji klasifikatorja spola na podlagi obraza uporabimo povezavo med metodo analize glavnih komponent [9, 26] in okoljem za podatkovno rudarjenje Orange [8]. Za učenje klasifikatorja uporabimo programsko knjižnico OpenCV [5] in del obsežne zbirke obrazov FERET [19, 20].

Naš sistem za digitalno karakterizacijo vsebuje tudi programske komponente, ki smo jih razvili za lažje in učinkovitejše praktično izkoriščanje pridobljenih podatkov. Namenjene so predvsem uporabi pri digitalnem oglaševanju. Kot primer navedimo trženje oglaševanja preko zaslonov na javnih površinah. Učinkovitost in tudi cena oglaševanja temeljita na številu ljudi, ki si ogledajo oglas. Običajno izhajajo iz pavšalnih, grobih ocen. Pavšalne ocene pa pri obračunavanju s pomočjo sistema za digitalno karakterizacijo na-

domestimo z dejanskimi, statistično obdelanimi in razčlenjenimi podatki, saj imamo popolno statistično informacijo ne samo o številu gledalcev, ampak celo o njihovih značilnostih. Podatke beležimo v podatkovni zbirki, kar nam omogoča kasnejše analize ter izdelavo poročil. Izdelali smo aplikacijo, ki omogoča pregledovanje zajetih podatkov preko spleta. Uporabljena podatkovna zbirka je PostgreSQL [21]. Operacijski sistem strežnika je Debian Linux [7], za prikaz spletnih strani pa uporabimo Apache [1]. Za dinamično pripravo zajetih podatkov uporabimo okolje PHP [18], za vizualizacijo pa tehnologijo Adobe Flex [4].

Pri razvoju našega sistema smo uporabili samo odprtokodno programsko opremo, ki je prosto dostopna na spletu. Sistem je zasnovan tako, da lahko teče na vseh množično zastopanih operacijskih sistemih (Windows, Linux, Mac OS) in zato lahko deluje na poljubnem računalniškem sistemu. Sistem smo tudi praktično testirali na video posnetkih znane turistične agencije in je zrel za komercialno uporabo na že obstoječih sistemih za digitalno oglaševanje. Prvo komercialno aplikacijo pričakujemo v poslovalnicah Pošte Slovenije.

Na podlagi tega dela smo objavili že nekaj znanstvenih publikacij [3, 22, 23, 24]. Raziskovalno in razvojno delo bomo nadaljevali predvsem v smeri natančnih in učinkovitih metod za določevanje detajlnih značilnosti obraza, zlasti starosti in razpoloženja. Pri tem je osnovni problem ta, da je metode treba razviti za aplikacijo na majhnih računalnikih in za delo v realnem času.

Poglavje 2

Digitalno oglaševanje in digitalna karakterizacija

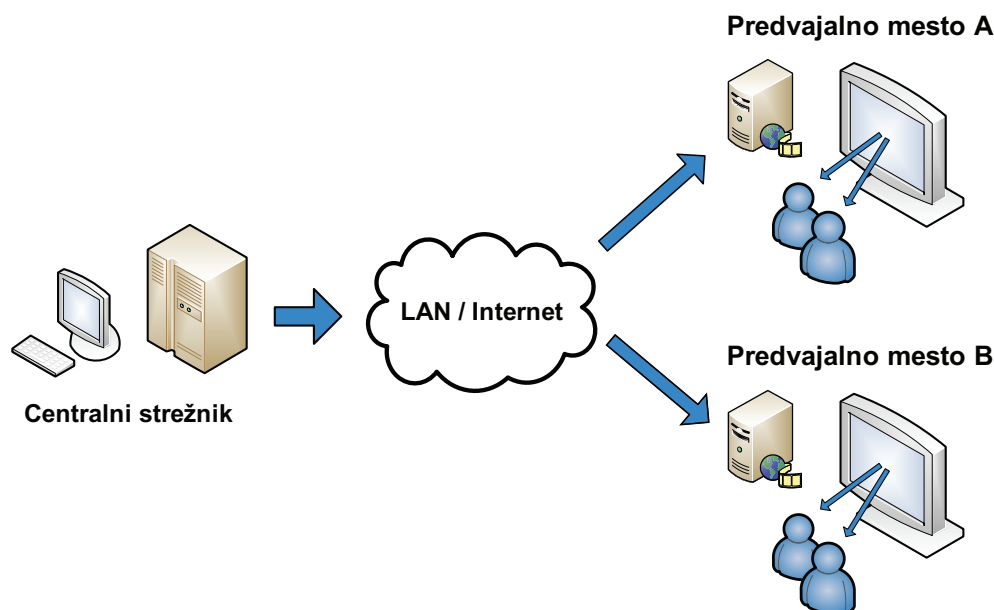
V poglavju opredelimo pojma *digitalno oglaševanje* in *digitalna karakterizacija*. Sistem za digitalno oglaševanje služi kot izhodišče za nadgradnjo in izboljšave, ki nas privedejo do lastnosti sistema za digitalno karakterizacijo.

2.1 Digitalno oglaševanje

Digitalno oglaševanje (angl. digital signage) je sistem za digitalno prikazovanje vsebine, ki se nahaja na javnih prostorih [15, 25]. Običajno ga sestavljajo osrednji strežnik in množica predvajalnih mest. Predvajalno mesto sestavlja zaslon in omrežno-predvajalni vmesnik, pogosto je to osebni računalnik. Povezavo med strežnikom in predvajalnimi mesti predstavlja komunikacijski kanal, ki navadno poteka po lokalnem omrežju (LAN) ali pa preko medmrežja. Centralni strežnik preko omrežja posreduje zahteve za prikaz vsebine na določenih predvajalnih mestih [11]. Shema sistema prikazuje slika 2.1.

Sistem za digitalno oglaševanje omogoča doseganje ciljev na podlagi upravljanja:

- zaporedja prikazanih vsebin,
- časovnega intervala trajanja posameznega prikaza,
- izbire vsebine za posamično predvajalno mesto,
- izbire vsebine glede na določen čas ali del dneva.



Slika 2.1: Shema sistema za digitalno oglaševanje. Komunikacija poteka od strežnika k predvajalnemu mestu.

Vrednotenje prikazane vsebine predstavlja velik problem sistemov za digitalno oglaševanje, saj temelji na ocenah in ne na trenutnih podatkih. Število ogledov vsebine se oceni zgolj na podlagi povprečnega števila ljudi, ki dnevno prečkajo območje predvajalnega mesta. Dolgoročno zastavljene, nekaj mesečne oglaševalske akcije tako dosežejo povprečje, pri kratkoročnih pa so nihanja velika. Tovrstne pavšalne ocene lahko precej odstopajo od povprečnih. Kot primer navedimo predvajalno mesto, ki se nahaja v nakupovalnem središču. Število ljudi, ki se bodo v času razprodaje nahajali v območju zaslona, bo verjetno večje od povprečnega, v času poletnih dopustov pa manjše. S pomočjo beleženja številka kupcev ne optimiziramo samo oglaševalske kampanje ampak lahko tudi prodajne kapacitete in strategijo.

V okviru vrednotenja omenimo še relevantnost prikazane vsebine. Predvajalno mesto na ukaz centralnega strežnika predvaja različne vsebine v določenih časovnih intervalih in določenem zaporedju. Učinkovitost podajanja vsebine lahko izboljšamo na podlagi karakterizacije opazovalcev. Zakaj ne bi skupini najstnic, če jih sistem prepozna kot najstnice, ponudili njim privlačne vsebine? Zakaj jim poleg relevantne vsebine sistem ne bi omogočili tudi neposredne interakcije?

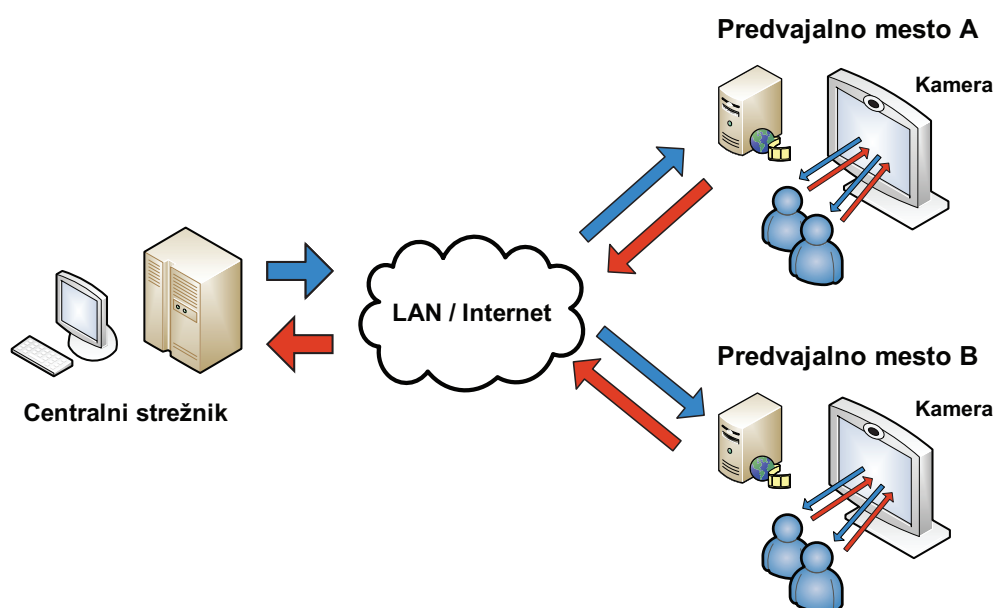
Digitalno oglaševanje ponuja precej možnosti izboljšav in razširitev. Inteli-

gentni sistem, ki ponuja odgovore na zgoraj zastavljene probleme, smo poimenovali sistem za digitalno karakterizacijo.

2.2 Digitalna karakterizacija

Digitalna karakterizacija je sistem za prikazovanje vsebin, ki omogoča sledenje in karakterizacijo gledalcev v realnem času, možnost interakcije z opazovalcem ter možnost analize zajetih podatkov s strani upravljavca [22, 23].

Postavitev sistema izhaja iz sistema za digitalno oglaševanje. Pri digitalni karakterizaciji predvajalna mesta opremimo z napravami za zajem slike, kot je na primer nizko cenovno kamera, vgrajena v okvir zaslona. Shemo sistema za digitalno karakterizacijo prikazuje slika 2.2.



Slika 2.2: Shema sistema za digitalno karakterizacijo. Z uporabo kamere omogočimo interakcijo med opazovalcem in sistemom. Komunikacija med strežnikom in predvajalnim mestom poteka v obe smeri.

V sistemu digitalne karakterizacije upravljaavec za vsako izmed predvajanih vsebin določi karakteristične lastnosti ciljne publike. Strežnik nato preko omrežja pošlje predvajalnemu mestu množico okarakteriziranih vsebin za predvajanje. Naloga predvajalnega mesta je, da na podlagi trenutnega dogajanja pred zaslonom karakterizira potencialne gledalce, izbere in prikaže najbolj rele-

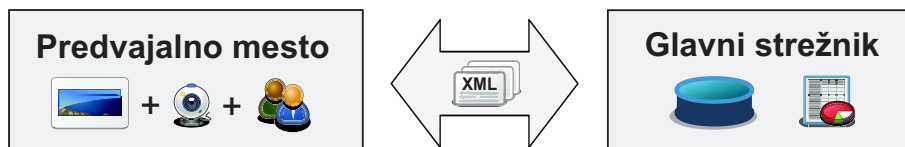
vantno vsebino ter spremlja njihov odziv. Karakterizacija gledalcev lahko zajema:

- določanje števila opazovalcev pred zaslonom,
- čas zadrževanje posameznega opazovalca,
- oceno oddaljenosti posameznega opazovalca od zaslona,
- klasifikacijo spola opazovalca na podlagi obraza,
- umestitev posameznega opazovalca v določeno starostno skupino,
- beleženje odziva opazovalca na prikazano vsebino.

Zajete karakteristike gledalcev se beležijo in omogočajo upravljavcu sistema pregled zgodovine dogajanja pred zaslonom. Na podlagi zajetih podatkov sistem omogoča pripravo statistik in generiranje poročil.

Varovanje osebnih podatkov, ki jih zajema sistem (slika osebe in vse, kar iz nje določimo) predstavlja poseben segment, ki je pomemben tako za uporabnike kot za izpolnjevanje zakonskih določil [28]. Že zasnova sistema za digitalno karakterizacijo mora zagotavljati, da zajem informacij obsega le kumulativne karakteristične podatke opazovalcev, ne pa tudi osebnih. Uporabimo zasnovo, ki minimizira možnost zlorabe osebnih podatkov, tako, da se sliko obdeluje lokalno, jo takoj po obdelavi zbriše in posreduje naprej v sistem le splošne statistične podatke.

Slika 2.3 prikazuje osnovni dve komponenti sistema za digitalno karakterizacijo: predvajalno mesto in glavni strežnik. Na predvajalnem mestu se zajema slika, na kateri želimo poiskati obraze različnih ljudi ter njihove karakteristike. Smiselno je, da karakterizacija poteka na predvajalnem mestu, ne pa na samem strežniku. To zagotavlja manjšo obremenitev omrežja ter predstavlja prvi nivo zagotavljanja zasebnosti, saj se namesto slike preko omrežja pošiljajo zgolj okarakterizirani podatki v obliki XML sporočil. Zajeto sliko zavržemo takoj po njeni obdelavi, saj se karakterizacija vrši v realnem času.



Slika 2.3: Osnovne komponente sistema za digitalno karakterizacijo.

Poglavje 3

Sledenje obrazov v realnem času

V poglavju predstavljamo opredelitev in opis realizacije prve komponente sistema, ki zajema detekcijo in sledenje opazovalcev na podlagi obraza.

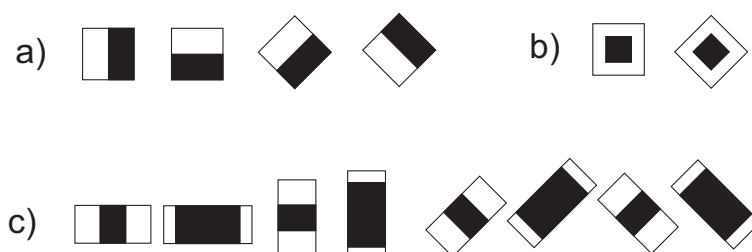
Procesiranje na predvajalnem mestu se opravlja v realnem času. To pomeni, da nas v vsakem trenutku zanima število ljudi in njihov položaj pred zaslonom. Predpostavljamo tudi, da na predvajalnem mestu nimamo visoko cenovne strojne opreme, temveč standardni osebni računalnik. Z upoštevanjem teh dveh zahtev ugotovimo, da bomo za detekcijo in sledenje obrazov potrebovali algoritem z majhno časovno kompleksnostjo. Metoda, ki zadostuje omenjenim kriterijem, je metoda detekcije obrazov avtorjev Viola in Jones, ki temelji na metodi strojnega učenja AdaBoost [14].

3.1 Detektor obrazov Viola in Jones

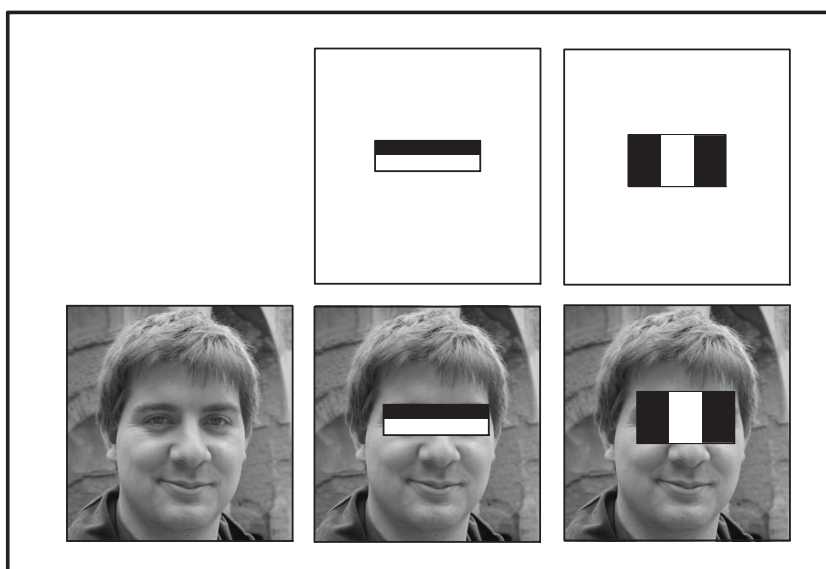
Leta 2001 sta Paul Viola in Michael Jones predlagala metodo za učinkovito realnočasno detekcijo obrazov [27]. Predlagani algoritem sestavljajo štirje temeljni koncepti:

- uvedba značilnic Haar, ki opisujejo preproste značilnice pravokotnih oblik,
- opredelitev integralne slike (angl. *integral image*) za hitro računanje značilnic,
- uporaba metode strojnega učenja AdaBoost za izgradnjo kaskade “močnih” klasifikatorjev,
- opredelitev zaporedja v kaskadi klasifikatorjev z namenom optimizacije izvajalnega časa.

Značilnice Haar (angl. Haar features) so osnovane na valčkih Haar (angl. Haar wavelets) [27]. Predstavljajo močan model za opis obraznega klasifikatorja ter omogočajo učinkovito implementacijo s pomočjo tehnike integriranja slike. Predstavljamo si jih lahko kot preproste pravokotne oblike črne in bele barve. Značilnice Haar v obliki maske sistematsko polagamo po sliki in ugotavljamo, katere in koliko slika vsebuje. Razširjen nabor značilnic Haar je predstavljen na sliki 3.1, primer ujemanja iskanih značilnic pri detekciji obraza pa na sliki 3.2.



Slika 3.1: Razširjen nabor značilnic Haar: a) robne značilnice, b) značilnice obkroženega središča (angl. center-surround features) in c) linijske značilnice.



Slika 3.2: Prvi dve značilnici Haar v osnovni Viola in Jones kaskadi za detekcijo obrazov iščeta oči.

Prisotnost značilnice ugotovimo z odštevanjem povprečne vrednosti temnega področja od povprečne vrednosti svetlega področja. Značilnica je prisotna, če razlika presega določen prag, ki se določi v fazi učenja klasifikatorja. Za računsko učinkovito preverjanje prisotnosti velikega števila značilnic Haar na sliki $I(x,y)$ sta Viola in Jones definirala integralno sliko $\Pi(x,y)$. Integralna slika $\Pi(x,y)$ v točki (x,y) vsebuje vsoto vrednosti vseh točk v področju nad in levo od točke (x,y) ter vrednosti $I(x,y)$ (glej sliko 3.3a). Definiramo jo kot:

$$\Pi(x, y) = \sum_{i=1}^x \sum_{j=1}^y I(i, j) \quad (3.1)$$

kjer $I(i,j)$ predstavlja vrednost v točki (i,j) osnovne slike I . Integralno sliko lahko izračunamo z enim samim prehodom slike na osnovi lastnosti:

$$S(x, y) = S(x, y - 1) + I(x, y), \quad (3.2)$$

$$\Pi(x, y) = \Pi(x - 1, y) + S(x, y), \quad (3.3)$$

kjer $S(x, y)$ predstavlja vsoto vrstice. Robna pogoja sta:

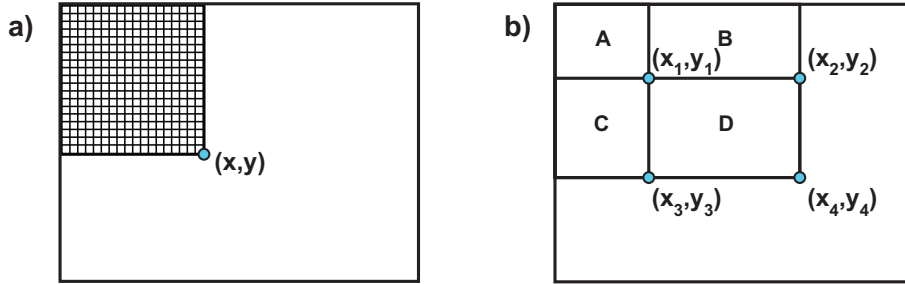
$$S(x, -1) = 0,$$

$$\Pi(-1, y) = 0.$$

Integralna slika nam omogoča računsko učinkovito računanje vsote področja pravokotne oblike. Za izračun vsote poljubnega pravokotnika potrebujemo zgolj 4 naslavljanja točk integralne slike in 3 računske operacije, kar predstavlja izreden prihranek pri računanju in umeščanju značilnic različnih velikosti in oblik [14]. Primer takšnega izračuna prikazuje slika 3.3b.

Za izračun povprečne sivinske vrednosti izračunano vsoto pravokotnika delimo z njegovo površino.

AdaBoost je meta algoritem, ki se velikokrat uporablja v povezavi z drugimi metodami strojnega učenja. Prvič sta ga opisala Yoav Freund in Robert Schapire [10]. Metoda AdaBoost uporabi množico “šibkih” (angl. weak) klasifikatorjev za izgradnjo “močnega” (angl. strong) klasifikatorja. Šibek klasifikator v tem primeru označuje klasifikator, katerega uspešnost detekcije mora presegati vsaj naključno ugibanje. Samostojna uporaba takšnega klasifikatorja ne daje iskanih rezultatov in ni smiselna. Predstavljamo pa si verigo šibkih klasifikatorjev, kjer vsak prispeva svoj delež do končnega odgovora. Če množico klasifikatorjev, zbranih v tej verigi, optimalno utežimo, dobimo močni klasifikator. Cilj algoritma AdaBoost je torej poiskati najboljšo kombinacijo



Slika 3.3: Integralna slika. a) Točka (x,y) integralne slike predstavlja vsoto vseh vrednosti osnovne slike znotraj označenega področja. b) Vsoto točk znotraj pravokotnika D lahko izračunamo kot $II(x_4, y_4) - II(x_2, y_2) - II(x_3, y_3) + II(x_1, y_1)$.

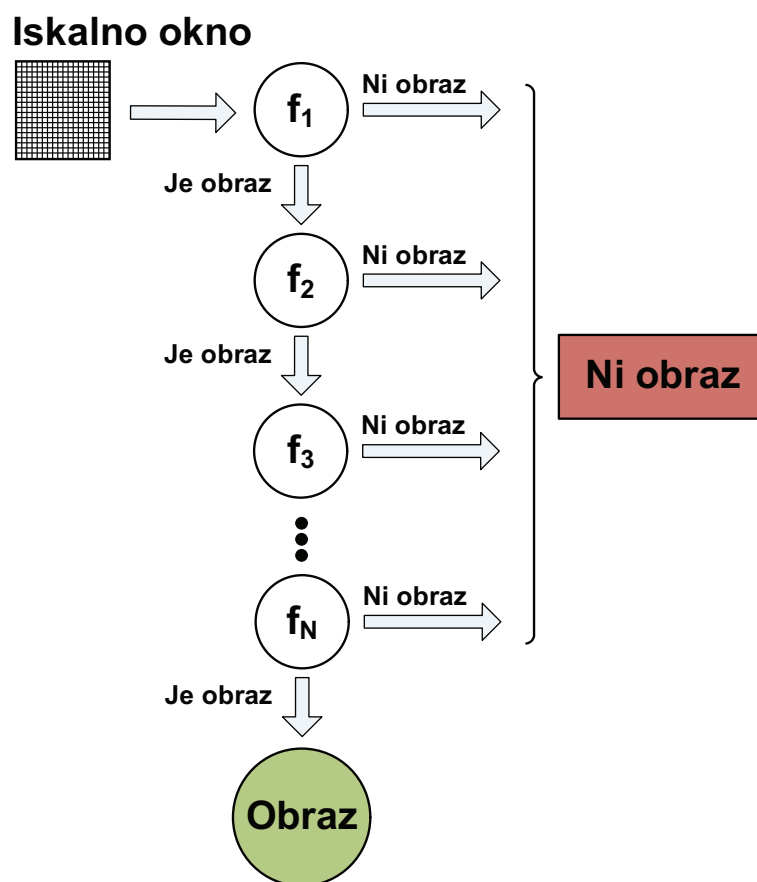
šibkih klasifikatorjev h_m in uteži α_m z namenom izgradnje močnega klasifikatorja H_M . Močni klasifikator H_M lahko formuliramo kot:

$$H_M(x) = \frac{\sum_{m=1}^M \alpha_m * h_m(x)}{\sum_{m=1}^M \alpha_m} \quad (3.4)$$

kjer x predstavlja vzorec za klasifikacijo, M pa število šibkih klasifikatorjev. Koeficienti α_m zavzemajo vrednosti večje ali enake 0, zalogo vrednosti $h_m(x)$ pa predstavljajo realna števila na intervalu $\{-1,1\}$. Linearno kombinacijo šibkih klasifikatorjev normaliziramo z vsoto uteži α_m [14].

Viola in Jones sta predlagala gradnjo šibkih klasifikatorjev na osnovi značilnic Haar. Iz množice dobljenih šibkih klasifikatorjev lahko sestavimo različne močne klasifikatorje. Med njimi iščemo takšno zaporedje, ki bo najbolj ustrezalo našemu problemu. Prag sprejetja močnega klasifikatorja mora biti nastavljen dovolj nizko, da sprejme vse oziroma skoraj vse učne primere. Ti navadno predstavljajo množico več 1000 slik. Zaporedje močnih klasifikatorjev imenujemo tudi kaskada klasifikatorjev. Primer kaskade za detekcijo obrazov prikazuje slika 3.4.

Omenimo, da lahko vrstni red že izbranih klasifikatorjev znatno vpliva na izvajalni čas detekcije. Smiselno je, da najbolj selektivne klasifikatorje uvrstimo na začetek kaskade, saj tako najhitreje zavrnemo največ primerov. Predlagana metoda detekcije je neodvisna od velikosti in lokacije obrazov na sliki, ne pa tudi rotacij obrazov.



Slika 3.4: Kaskada klasifikatorjev. Del slike, ki je prešel skozi celotno kaskado označimo za obraz, sicer ga zavrnamo.

3.2 Selekcija in validacija obrazov

Metoda detekcije Viola in Jones se izkaže za učinkovito pri hitrem iskanju obrazov na sliki. Praktična uporaba metode pa pokaže na njeno pomanjkljivost pri natančnosti detekcije. S spreminjanjem praga detekcije lahko določimo, kako “strog” naj bo detektor pri iskanju obrazov.

Pri nizkem pragu detektor na enaki sliki detektira večje število obrazov, saj kriteriji za detekcijo niso več tako zelo strogi. Poveča se število pozitivnih detekcij, nekatere obraze detektira celo večkrat. Poleg pozitivnih detekcij pa se poveča tudi število negativnih detekcij, torej delov slike, ki v resnici ne predstavljajo obrazov. Visok prag nas pripelje do bolj natančne detekcije, saj se izognemo napačnim detekcijam in večkratnim detekcijam istega obraza,

vendar s tem izgubimo tudi detekcijo marsikaterega pravega obraza. Naš cilj je izboljšati detektor, da bo našel kar največje število pravih obrazov s čim manjšim številom napačnih detekcij. Predlagamo postopek, ki maksimizira pozitivne detekcije in minimizira negativne.

Ker želimo detektirati kar največ obrazov, izhajamo z detektorjem obrazov, ki ima ohlapne kriterije detekcije. Iz množice kandidatov moramo izločiti večkratno detektirane obraze in detekcije, ki ne predstavljajo obrazov. Prvi problem rešimo z združevanjem večkratnih detekcij istega obraza. Ideja temelji na tem, da preverimo ali se večji del dveh detektiranih obrazov prekriva. Ko najdemo tak par obrazov, obraza združimo (tj. povprečimo položaj in velikost) v nov obraz in osnovna obraza zavržemo. Primer psevdo kode podaja algoritem 1, rezultat združevanja pa slika 3.5.

Algoritem 1 Združevanje večkratnih detekcij istega obraza

Vhod: Seznam koordinat detektiranih obrazov v danem trenutku $NewFaces = \{F_1, F_2, \dots, F_N\}$, dolžina seznama N ,

Parametri: Faktor prekrivanja $\varepsilon_{overlap}$

Inicializacija: Seznam $MergedFaces = \{\}$

for $i = 1, \dots, N$ **do**

$CurrFace \leftarrow F_i | F_i \in NewFaces$

for $j = i + 1, \dots, N$ **do**

$SuccFace = F_j | F_j \in NewFaces \wedge F_j \neq F_i$

if ($overlapRatio(CurrFace, SuccFace) > \varepsilon_{overlap}$) **then**

$CurrFace \leftarrow merge(CurrFace, SuccFace)$

$replaceListElement(NewFaces, F_i, CurrFace)$

$discardListElement(NewFaces, F_j)$

$N \leftarrow N - 1$

end if

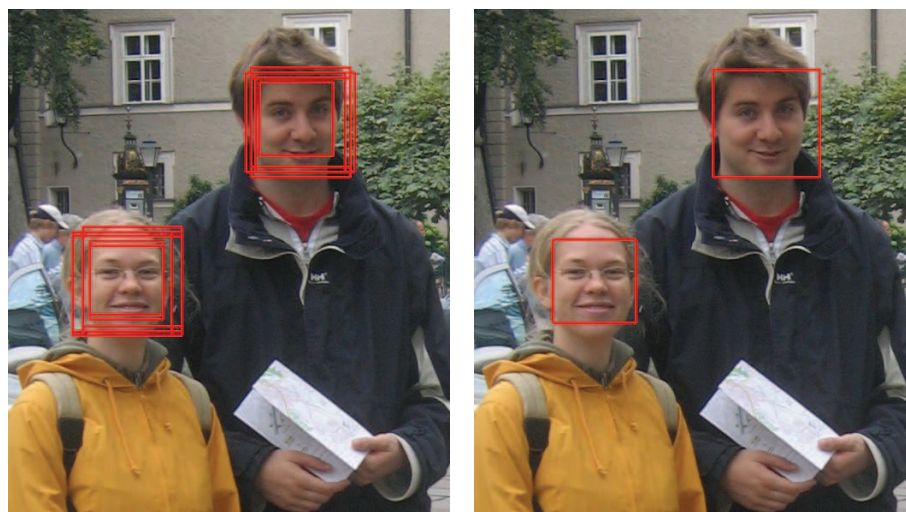
end for

end for

$MergedFaces \leftarrow NewFaces$

Izhod: Seznam združenih obrazov $MergedFaces$

Naslednji problem predstavljajo napačne detekcije. Praktična uporaba detektorja obrazov Viola in Jones pokaže, da se napačne detekcije pojavljajo sporadično. Ob ohlapnih kriterijih bo detektor v daljšem časovnem intervalu od časa do časa in navadno na različnih krajih javil napačno detekcijo. Dejstvo, da nas zanima dogajanje v določenem časovnem intervalu in ne zgolj ena sama slika ter skoraj naključno pojavljanje napačnih detekcij, nam bo koristilo



Slika 3.5: Združevanje večkratnih detekcij istega obraza.

pri njihovi identifikaciji.

Predlagamo metodo, ki validira detektirane obraze na podlagi prostorske in časovne lokalnosti. Detektiran obraz se najprej označi kot kandidat za dejanski obraz. Če je bila znotraj krajšega časovnega intervala $t_{discard}$ na območju obraza uspešna detekcija vsaj D -krat, se obraz označi kot veljaven obraz. V nasprotnem primeru se obraz zavrže, saj se predvideva, da gre za primer negativne detekcije. Primer psevdo kode predstavlja algoritem 2.

Praktična uporaba zgoraj opisanih algoritmov pokaže, da smo uspešno rešili problem ohlapnega detektorja obrazov. Dopolnjeni detektor uspešno zavrača sporadične negativne detekcije osnovnega detektorja Viola in Jones ter omogoča detekcijo obrazov, ki jih ob strožjih kriterijih detekcije ne bi našli.

Algoritem 2 Validacija detektiranih obrazov

Vhod: Seznam detektiranih obrazov v danem trenutku $NewFaces = \{F_1, F_2, \dots, F_N\}$, seznam kandidatov za obraze $CandFaces = \{C_1, C_2, \dots, C_M\}$, seznam aktivnih obrazov $ValidFaces = \{V_1, V_2, \dots, V_K\}$

Parametri: Faktor prekrivanja $\varepsilon_{overlap}$, število detekcij D , čas izločitve $t_{discard}$

```

for all  $F_i$  in  $NewFaces$  do
   $CurrFace \leftarrow F_i$ 
   $FaceFound \leftarrow \text{false}$ 
  for all  $C_j$  in  $CandFaces$  do
     $CandFace \leftarrow C_j$ 
    if  $CandFace_{appearanceTime} > t_{discard}$  then
       $\text{discardListElement}(CandFaces, C_j)$ 
    else if  $(\text{overlapRatio}(CurrFace, CandFace) > \varepsilon_{overlap})$  then
       $CurrFace_{timesDetected} \leftarrow CurrFace_{timesDetected} + 1$ 
       $\text{discardListElement}(NewFaces, F_i)$ 
       $FaceFound \leftarrow \text{true}$ 
    end if
  end for
  if  $FaceFound \neq \text{true}$  then
     $CurrFace_{timesDetected} \leftarrow 1$ 
     $\text{addElementToList}(CandFaces, F_i)$ 
  end if
end for
for all  $C_i$  in  $CandFaces$  do
   $CurrCand \leftarrow C_i$ 
  if  $CurrCand_{timesDetected} > D$  then
     $\text{moveElement}(C_i, CandFaces, ValidFaces)$ 
  end if
end for
Izhod: Posodobljena seznama  $CandFaces$  in  $ValidFaces$ .

```

3.3 Sledenje obrazov v realnem času

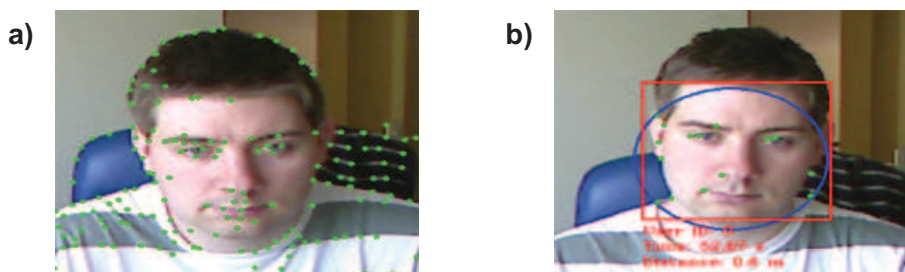
Opišimo, kako nepovezano detektiranje obrazov spremeniti v sistem za sledenje. Metoda sledenja obrazov predstavlja časovno in prostorsko povezovanje detektiranih obrazov. Zanima nas torej čas pojavljanja in gibanje posameznega obraza. Detektor obrazov Viola in Jones pri svojem delovanju upošteva zgolj trenutno sliko, na kateri se izvaja detekcija. Prva ideja povezovanja zaporednih detekcij izhaja kar iz algoritma 2 za validacijo detektiranih obrazov. Upošteva joč časovno in prostorsko lokalnost lahko predpostavimo, da obraza, ki sta bila detektirana zgolj nekaj desetink sekunde narazen na skoraj enakem delu slike, pripadata isti osebi. Preprost sistem sledenja torej vsebuje seznam trenutno aktivnih obrazov ter za vsakega čas in položaj ob zadnji detekciji. Obraz ostane v seznamu aktivnih obrazov toliko časa, dokler detektor ob vnovični detekciji ne poišče obraza, ki se nahaja na podobnem mestu. Če v določenem časovnem intervalu obraz ni ponovno zaznan, postane neaktiven in se posledično briše iz seznama aktivnih obrazov.

Poleg metode združevanja zaporedno detektiranih obrazov smo sledenju obrazov dodali še metodo optičnega toka (angl. optical flow). Optični tok se uporablja za določanje gibanja na podlagi dveh časovno zaporednih slik. Na osnovi prejšnje slike za vsako točko trenutne slike ocenimo smer in hitrost gibanja. Splošna metoda optičnega toka je računsko intenzivna, zato smo uporabili metodo, ki računa optični tok zgolj za izbrano množico ugodnih točk. Ugodne točke so tiste, ki ležijo na robovih ploskev oziroma predstavljajo kot, kar omogoča hitro, zanesljivo in robustno sledenje. Metodo sledenja ugodnih točk poznamo tudi pod imenom avtorjev te metode kot metodo Lucas-Kanade [5]. Temelji na treh predpostavkah:

- Konsistentna osvetljenost. Sivinska oziroma barvna vrednost določene točke skozi čas ostaja enaka.
- Časovna lokalnost. Točke med dvema zaporednima slikama se ne premikajo hitro.
- Prostorska koherenca. Področje točk, ki pripada istemu delu scene, se obnaša in giblje enako.

Vsak detektiran obraz torej opišemo še z množico ugodnih točk, ki so znotraj oziroma v neposredni bližini obraza. Zanimivo je, da se pogosto za najbolj ugodne točke obraza izkažejo ravno oči, usta ter konica nosu. Problem sledenja obraza smo torej prevedli na problem sledenja ugodnih točk. Za vsako zajeto

sliko izračunamo optični tok za izbrane točke, ki nam povedo kam se je premaknil obraz. Primer delovanja metode prikazuje slika 3.6.



Slika 3.6: Metoda optičnega toka Lucas-Kanade. a) Ugodne točke za sledenje znotraj in v okolici obraza. b) Izračun novih koordinat obraza na podlagi sledenja točk. Obraz določen na osnovi optičnega toka predstavlja modra elipsa.

3.4 Ocena razdalje

Zanimiv podatek sistema za digitalno karakterizacijo je, kako daleč od zaslona za projiciranje vsebine se nahaja posamezen opazovalec. V našem primeru predpostavljamo, da je kamera, ki zajema sliko predvajalnega mesta, vgrajena nad zaslon (glej sliko 2.2). Problem razdalje med opazovalcem in zaslonom tako prevedemo na problem razdalje med opazovalcem in kamero.

Za natančen izračun razdalje med obrazom in zaslonom bi potrebovali dve kalibrirani kameri, ki bi opazovali predvajalno mesto. Sistem za digitalno karakterizacijo je bolj preprost in vključuje zgolj eno kamero. Predpostavka, da se fizične velikosti obrazov med seboj bistveno ne razlikujejo, nam omogoča oceno oddaljenosti posameznega obraza od zaslona. Ko detektor na zajeti sliki zazna obraz, lahko izračunamo velikost obraza kot njegovo ploščino P na sliki. Ploščina P je povezana z oddaljenostjo obraza, saj izgledajo objekti bližje kameri večji, kot objekti, ki so daleč od kamere. Na podlagi te ugotovitve empirično določimo korelacijsko tabelo, ki povezuje ploščino obraza z razdaljo od kamere. Korelacijsko tabelo prikazuje tabela 3.1. Velja opozoriti, da so podatki v tabeli 3.1 izmerjeni na specifični kameri z vidnim kotom 72° za točno specifičen obraz in smiselno zaokroženi.

Z interpolacijo tabele lahko izrazimo obliko funkcije za oceno razdalje $dist$ na osnovi ploščine obraza P . Predpostavimo, da v prvi aproksimaciji obliko obraza lahko aproksimiramo s kvadratom. Stranico kvadrata izračunamo kot

Ploščina [px]*	Koren ploščine	Razdalja do kamere [m]
15000	122,5	0.8
9000	94,9	0.85
5000	70,7	1.6
2500	50,0	2.1

*točkovni element slike (angl. pixel)

Tabela 3.1: Empirična korelacijska tabela, ki povezuje ploščino obraza na sliki z razdaljo do kamere.

koren ploščine. Detektor je pri določanju višine in dolžine obraza na sliki podvržen določeni napaki. Pri izračunu ploščine se ta napaka še poveča. Napako zmanjšamo, če ploščino korenimo, ter za "osnovno" enoto velikosti obraza vzamemo kar stranico kvadrata. Iz tabele 3.1 razberemo, da sta razdalja in velikost približno obratno sorazmerni in funkcijo *dist* zapišemo kot:

$$dist(P) = \frac{A}{\sqrt{P}} \quad (3.5)$$

kjer je A številska konstanta, ki jo dobimo iz korelacije v tabeli 3.1. Predlagana metoda kljub grobim predpostavkam deluje dobro, saj pri ocenjevanju razdalje dosega med 5 do 10% odstopanja od izmerjene vrednosti. Tovrstna metoda ima tudi izredno nizko časovno kompleksnost. Večjo natančnost lahko dosežemo z bolj kompleksnim modelom funkcije, kar pa vpliva na večjo časovno zahtevnost.

Praktična uporaba metode na osnovi podatkov zajetih v realnem času z detektorjem Viola in Jones pokaže nov problem. Tudi če gledalec na predvajalnem mestu miruje, detektor njegov obraz ob vsaki novi detekciji zazna nekoliko različno, kar povzroči, da se pojavijo pozitivni in negativni odkloni v ocenjeni razdalji. Problem smo rešili s povprečenjem zadnjih N ocenjenih razdalj in uvedbo funkcije povprečene razdalje *distAvg*:

$$distAvg(P_t) = \frac{dist(P_t) + \sum_{i=1}^N dist(P_{t-i})}{N+1} \quad (3.6)$$

kjer P_t predstavlja ploščino obraza v času t . Ob praktični uporabi se izkaže, da povprečenje reši problem odklanjanja razdalje že za zadnjih 8 slik ($N = 8$).

3.5 Implementacija

Pri opredelitvi okolja za realizacijo sistema za sledenje obrazov smo izhajali iz sledečih zahtev:

- okolje omogoča hitro izvajanje programske kode, ki dopušča manipulacijo na relativno nizkem nivoju (upravljanje s pomnilnikom),
- možnost uporabe primerne zbirke metod računalniškega vida, ki bi služila kot osnova pri implementaciji sistema za detekcijo in sledenje obrazov,
- čim večja neodvisnost programskega okolja in knjižnic od operacijskega sistema in strojne opreme,
- okolje mora ponujati možnosti za skalabilno realizacijo.

Pomemben parameter pri izbiri programskega okolja je tudi izvajalni čas programa. Želimo tudi, da pri implementaciji lahko neposredno manipuliramo s pomnilnikom, saj s tem pridobimo boljši nadzor nad njegovo porabo. Odločili smo se za programsko okolje zasnovano v programskem jeziku C++, saj nadgrajuje konstrukte jezika C in omogoča lažje delo z objekti.

3.5.1 Programska knjižnica OpenCV

Pri izboru programskih knjižnic računalniškega vida smo se odločili za programski paket OpenCV. OpenCV je odprtokodna zbirka metod računalniškega vida, napisana v programskih jezikih C in C++. Knjižnica je rezultat dela razvojnega oddelka podjetja Intel. Namenjena je široki uporabi, saj je izdana pod licenco BSD, kar omogoča prosto uporabo tako za raziskovalne kot tudi komercialne aplikacije. Primarno je namenjena procesiranju slik v realnem času [5]. Zbirka OpenCV ponuja poenoten vmesnik do različnih naprav za zajemanje slik ter manipulacijo z njimi. Vključuje tudi detektor Viola in Jones ter učinkovito implementacijo metode za izračun optičnega pretoka Lucas-Kanade. Dodatno prednost predstavlja sama zasnova knjižnice, saj njeno delovanje ni omejeno na izbiro operacijskega sistema in strojne opreme.

3.5.2 Programska knjižnica Boost

Boost je zbirka pregledanih (angl. peer reviewed) odprtokodnih in prosto dostopnih knjižnic, ki razširjajo funkcionalnosti jezika C++ [16]. Cilji projekta Boost so vključitev čim večjega števila novih knjižnic v standard jezika

C++. Nabor knjižnic je širok, saj zajema tako splošno namenske, kot tudi specifične in abstrakcijske knjižnice. Več kot 80 zbranih knjižnic vključuje metode iz področja linearne algebre, regularnih izrazov, nitenja, (angl. *threading*), generiranja psevdo-naključnih števil in še mnogo drugih. Vse knjižnice so zasnovane neodvisno od platforme.

Pomembno lastnost sistema predstavlja skalabilnost, ki smo jo dosegli z uporabo nitenja. Večnitna aplikacija omogoča vzporedno izvajanje različnih komponent programa, saj se posamezne komponente razporedijo po različnih procesorjih in jedrih. Sočasno izvajanje lahko bistveno zmanjša izvajalni čas, ki predstavlja kritični element aplikacije. Prednost delitve programa med več niti predstavlja tudi lažje upravljanje z izvorno kodo, saj posamezna nit zajema logično zaključeno enoto.

Pri implementaciji večnitne aplikacije smo se oprli na knjižnico znotraj programskega paketa Boost. Program smo razdelili na tri logične enote:

- *FaceTracker*, ki skrbi za sledenje obrazov,
- *FaceClassifier*, ki omogoča karakterizacijo obraza,
- *DataTracker*, ki omogoča beleženje zajetih podatkov,

kjer vsaka enota predstavlja svojo nit.

Kumulativni podatki se beležijo v formatu XML. Hranijo se v datoteki na lokalnem datotečnem sistemu in tudi v podatkovni zbirki na centralnem strežniku. Za pošiljanje podatkov preko omrežja smo uporabili knjižnico Asio, ki združuje zbirko nizkonivojskih metod za mrežno programiranje [2]. Zajeti podatki se v realnem času preko protokola HTTP posredujejo na glavni strežnik.

Poglavje 4

Karakterizacija obrazov

Glavno lastnost sistema za digitalno karakterizacijo predstavlja njegova sposobnost realnočasne karakterizacije opazovalcev na predvajanem mestu. S tem dosežemo večjo relevantnost predvajane vsebine ter upravljanje na osnovi realnih podatkov. V poglavju 3 smo predstavili del sistema, ki skrbi za detekcijo in sledenje obrazov, sedaj pa se bomo osredotočili na karakterizacijo značilnic posameznega obraza.

Na podlagi obraza lahko določimo lastnosti opazovalca [22], kot so:

- spol,
- starostni razred,
- rasna skupina,
- razpoloženje na podlagi izraza in mimike,
- obrazne modalnosti, kot so brada, brki, očala, pokrivala.

Za razpoznavanje omenjenih lastnosti je bilo razvitih že več metod računalniškega vida. Metode so večinoma specifične, zato predlagamo postopek karakterizacije, ki se lahko uporabi za določanje širokega nabora značilnic. V delu se bomo omejili zgolj na klasifikator za določanje spola, kljub temu da bi enak postopek lahko uporabili tudi za določanje starostne in rasne skupine opazovalcev. Postopek temelji na povezavi metode glavnih komponent (angl. Principal Component Analysis oz. PCA) s programom za podatkovno rudarjenje Orange.

4.1 Metoda glavnih komponent

Metoda glavnih komponent (angl. PCA) določi ortogonalno bazo lastnih vektorjev - slik, s katerimi lahko najboljše popišemo množico vhodnih slik in pri tem izgubimo čim manj informacije. Namen metode je zmanjšanje števila dimenzij mnogo dimenzijskega problema na osnovi analize glavnih komponent. Rezultati metode so abstrahirane značilnice, ki predstavljajo učno množico v nižje dimenzijskem lastnem prostoru (angl. eigenspace). Metoda minimizira napako rekonstrukcije slike ter maksimizira varianco projekcije v prostoru lastnih vektorjev [9, 14, 26].

Prvi del metode PCA je priprava učne množice. Število slik v učni množici naj bo enako N . Vse vhodne slike poravnamo in pretvorimo na enako velikost $(m \times n)$. Sledi normalizacija barvne oziroma sivinske intenzitete, kar pomeni, da za vsako vhodno sliko I_j velja $\sum_{u=1}^m \sum_{v=1}^n I(u, v)^2 = 1$. Vsako vhodno sliko I_j velikosti $(m \times n)$ iz učne množice pretvorimo v vektor \mathbf{x}_j velikosti $(m \cdot n \times 1)$. Pretvorba dvodimenzionalne slike v enodimenzionalen vektor poteka tako, da beremo elemente slike od leve proti desni, od zgoraj navzdol in jih zaporedoma dodajamo v vektor. Vektor \mathbf{x}_j lahko formaliziramo kot:

$$\mathbf{x}_j = [I_{11}, I_{12}, \dots, I_{1n}, I_{21}, \dots, I_{mn}]^T, \quad (4.1)$$

kjer I_{uv} predstavlja element slike $I_j(u, v)$. Iz N dobljenih vektorjev izračunamo povprečni vektor $\bar{\mathbf{x}}$:

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{j=1}^N \mathbf{x}_j, \quad (4.2)$$

kjer $\bar{\mathbf{x}}$ predstavlja povprečno sliko učne množice v vektorski obliki. Sedaj lahko zapišemo matriko \mathbf{X} dimenzije $[m \cdot n \times N]$ kot:

$$\mathbf{X} = [(\mathbf{x}_1 - \bar{\mathbf{x}}) | (\mathbf{x}_2 - \bar{\mathbf{x}}) | (\mathbf{x}_3 - \bar{\mathbf{x}}) | \dots | (\mathbf{x}_N - \bar{\mathbf{x}})], \quad (4.3)$$

kjer vektorji \mathbf{x}_j predstavljajo slike učne množice v vektorski obliki in znak $|$ pomeni, da vektorje v matriko zlagamo po stolpcih. Matriko \mathbf{X} uporabimo za izračun kovariančne matrike \mathbf{Q} , ki nam pove, v kakšni medsebojni odvisnosti so vhodne slike. Matriko \mathbf{Q} definiramo kot:

$$\mathbf{Q} = \mathbf{X}\mathbf{X}^T, \quad (4.4)$$

kjer \mathbf{Q} predstavlja kovariančno matriko velikosti $(m \cdot n \times m \cdot n)$. Matrika \mathbf{Q} je diagonalno simetrična kvadratna matrika. Pri izračunu lastnih vektorjev in

lastnih vrednosti matrike \mathbf{Q} uporabimo metodo razcepa na singularne vrednosti (angl. singular value decomposition oziroma SVD). Definirajmo matriko $\hat{\mathbf{Q}}$ kot:

$$\hat{\mathbf{Q}} = \mathbf{X}^T \mathbf{X}, \quad (4.5)$$

kjer je $\hat{\mathbf{Q}}$ diagonalno simetrična kvadratna matrika velikosti $(N \times N)$. Matriki \mathbf{Q} in $\hat{\mathbf{Q}}$ imata enake lastne vrednosti [26], kar nam omogoča, da lahko lastne vektorje matrike \mathbf{Q} izračunamo iz lastnih vektorjev matrike $\hat{\mathbf{Q}}$. Velja torej:

$$\lambda = \hat{\lambda}, \quad (4.6)$$

$$\boldsymbol{\nu} = \mathbf{X} \cdot \hat{\boldsymbol{\nu}}, \quad (4.7)$$

kjer je λ lastna vrednost matrike \mathbf{Q} , $\boldsymbol{\nu}$ lastni vektor matrike \mathbf{Q} , $\hat{\lambda}$ lastna vrednost matrike $\hat{\mathbf{Q}}$ in $\hat{\boldsymbol{\nu}}$ lastni vektor matrike $\hat{\mathbf{Q}}$. Tipično je $N \ll m \cdot n$, zato je izračun lastnih vektorjev in lastnega sistema matrike $\hat{\mathbf{Q}}$ numerično ugodnejši.

Lastni vektorji matrike $\hat{\mathbf{Q}}$ so netrivialne rešitve ($\hat{\boldsymbol{\nu}} \neq \mathbf{0}$) karakteristične enačbe:

$$(\hat{\mathbf{Q}} - \hat{\lambda} \cdot \mathbf{I}) \cdot \hat{\boldsymbol{\nu}} = 0, \quad (4.8)$$

kjer je \mathbf{I} enotska matrika [9, 26]. Ker je $\hat{\mathbf{Q}}$ matrika reda $(N \times N)$, ima N lastnih vrednosti in N lastnih vektorjev $\hat{\boldsymbol{\nu}}_i$. Na osnovi vektorjev $\hat{\boldsymbol{\nu}}_i$ po enačbi 4.7 izračunamo vektorje $\boldsymbol{\nu}_i$.

Za vsak slikovni vektor \mathbf{x}_j izračunamo njegove projekcije g_{ji} na lastne vektorje $\boldsymbol{\nu}_i$ kot:

$$g_{ji} = \boldsymbol{\nu}_i^T \cdot (\mathbf{x}_j - \bar{\mathbf{x}}). \quad (4.9)$$

Projekcije uredimo v vektor \mathbf{g}_j dolžine N , katerega zapišemo kot:

$$\mathbf{g}_j = [g_{j1}, g_{j2}, \dots, g_{jN}]. \quad (4.10)$$

Vhodno sliko I_j lahko sedaj rekonstruiramo v vektorski obliki \mathbf{x}_j kot:

$$\mathbf{x}_j = \bar{\mathbf{x}} + \sum_{i=1}^N g_{ji} \boldsymbol{\nu}_i, \quad (4.11)$$

kjer $\bar{\mathbf{x}}$ predstavlja povprečno sliko sistema, g_{ji} i -to komponento projekcije \mathbf{g}_j in $\boldsymbol{\nu}_1, \dots, \boldsymbol{\nu}_N$ lastne vektorje kovariančne matrike \mathbf{Q} .

Izračunane pare lastnih vrednosti in lastnih vektorjev nato uredimo padajoče glede na lastno vrednost, tako da velja $\lambda_1 \geq \lambda_2 \geq \dots \lambda_N$. Pari z večjo lastno

vrednostjo nosijo več informacije, saj lastna vrednost v tem primeru predstavlja varianco od povprečne slike $\bar{\mathbf{x}}$ [14]. Tipično je $\lambda_i \approx 0$, za $i > K$, kjer je $K \ll N$. Pare z manjšo varianco kot λ_i običajno zanemarimo, saj se v praksi pokaže, da je $K \ll N$, kar pomeni, da prostor slik, ki je sprva vseboval nekaj tisoč dimenzij, zmanjšamo skoraj za dve dekadi velikosti. Kljub redukciji obdržimo večino informacije za rekonstrukcijo slik, saj velja:

$$\mathbf{x}_j \approx \bar{\mathbf{x}} + \sum_{i=1}^K g_{ji} \boldsymbol{\nu}_i. \quad (4.12)$$

Pri iskanju korelacije med novo sliko I_r in tistimi v učni množici si lahko pomagamo z izračunanimi projekcijami slik učne množice. Korelacijo v slikovnem prostoru nadomestimo z evklidsko razdaljo dveh vektorjev v lastnem prostoru sistema [26]. Namesto zamudnega množenja vektoriziranih slik med seboj, sliko I_r pretvorimo v vektor \mathbf{x}_r (enačba 4.1) in izračunamo projekcijo na lastne vektorje sistema \mathbf{g}_r (enačbi 4.9 in 4.10). Nato za vsak projekcijski vektor učne množice \mathbf{g}_j izračunamo evklidsko razdaljo do \mathbf{g}_r kot:

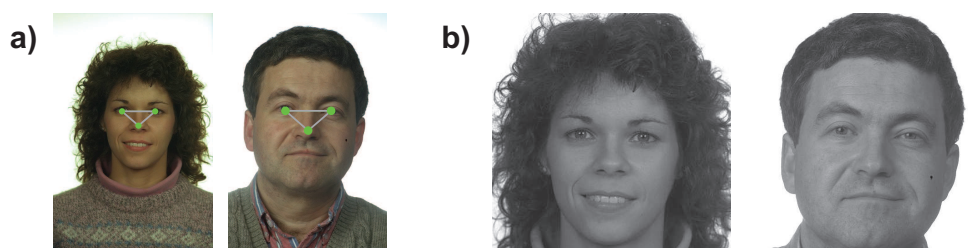
$$d(\mathbf{g}_j, \mathbf{g}_r) = \|\mathbf{g}_j - \mathbf{g}_r\| = \sqrt{\sum_{m=1}^N (g_{jm} - g_{rm})^2}, \quad (4.13)$$

kjer g_{jm} predstavlja j -to komponento vektorja \mathbf{g}_j , g_{rm} pa j -to komponento vektorja \mathbf{g}_r . Primerjava evklidskih razdalj za določanje korelacije močno zmanjša časovno kompleksnost iskanja podobnosti med slikami.

4.2 Klasifikacija spola

4.2.1 Projekcija slik z metodo glavnih komponent

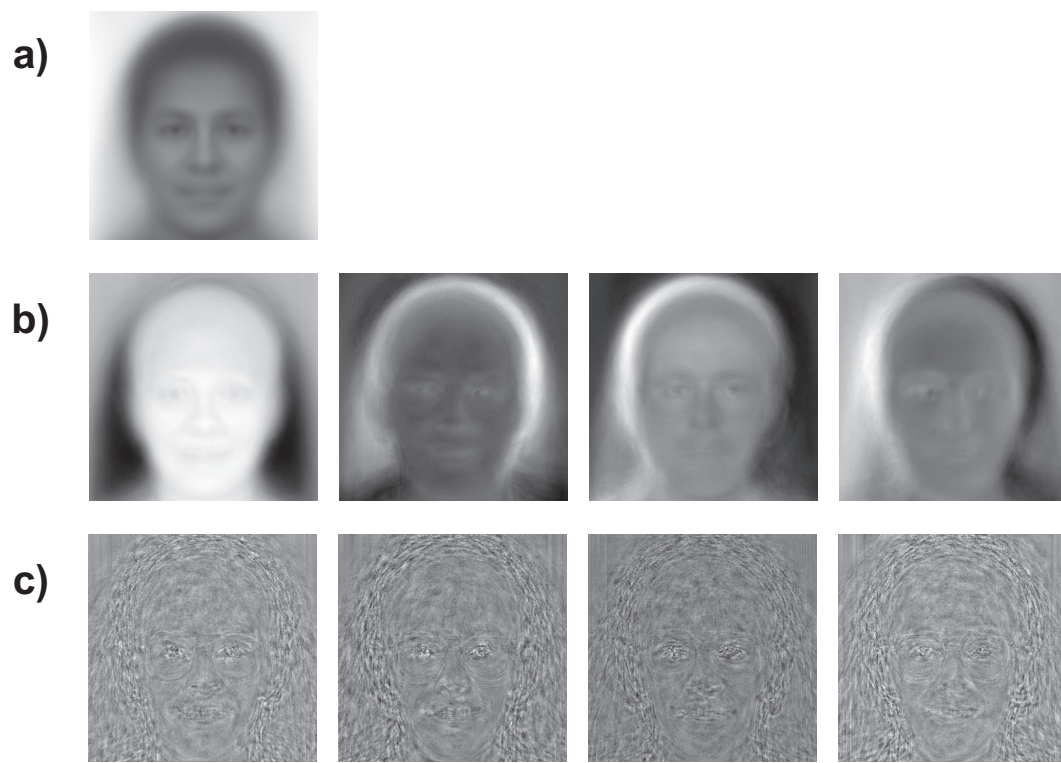
Prvi del sistema za klasifikacijo spola zajema pripravo vhodne množice. Uporabimo zbirko FERET, ki vsebuje 2.413 barvnih slik 856 različnih oseb [19, 20]. Zbirka je namenjena predvsem raziskovalcem na področju računalniškega vida, ki se ukvarjajo z razpoznavanjem ljudi na osnovi obrazov. Vsaka oseba je fotografirana z različnimi modalnostmi (očala, brada) v različnih časovnih obdobjih. Za potrebe učenja smo iz zbirke naključno izbrali 400 slik ženskih in 400 slik moških obrazov. Struktura slik zajema različne starostne razrede in rasne skupine. Izbrane slike nato normaliziramo. Predprocesiranje tako zajema poravnavo slik na osnovi ročnega določanja koordinat oči ter nosu. Razdaljo med koordinatami oči in nosu uporabimo za določevanje območja obraza. Obrezano in rotirano sliko, velikosti (400×400) točk, nato pretvorimo na sivinsko skalo. Primer predprocesiranja prikazuje slika 4.1.



Slika 4.1: Normalizacija vhodne množice. a) Sliki z označenimi očmi in nosom iz zbirke FERET. b) Sliki po končanem predprocesiranju.

Slike iz vhodne množice pretvorimo v vrstično obliko na podlagi enačbe 4.1 ter jih združimo v skupno matriko \mathbf{X} (enačba 4.3). Po izračunu povprečne slike, kovariančne matrike, njenih lastnih vektorjev in lastnih vrednosti izračunamo še lastne obraze (angl. eigenfaces). Lastni obrazi so po definiciji lastni vektorji $\boldsymbol{\nu}_i$ sistema (enačba 4.8), pretvorjeni v slike. Postopek pretvorbe vektorja v sliko je inverzen postopku, ki ga opisuje enačba 4.1. Primer povprečne slike in lastnih obrazov prikazuje slika 4.2.

Izračunamo tudi projekcije vhodnih slik na lasten sistem in jih zabeležimo. Pri shranjevanju smo ohranili vse projekcijske dimenzije ($K = N$), saj redukcija ni smiselna, ker želimo v fazo strojnega učenja prenesti kar največ informacije o problemu. Za množenje vektORIZIRANE slike z 800 lastnimi vektorji sistema potrebujemo 555,1 ms oziroma 0,7 ms na vektor. Čas je izmerjen na osebnem računalniku s frekvenco procesorja 2,14 GHz.



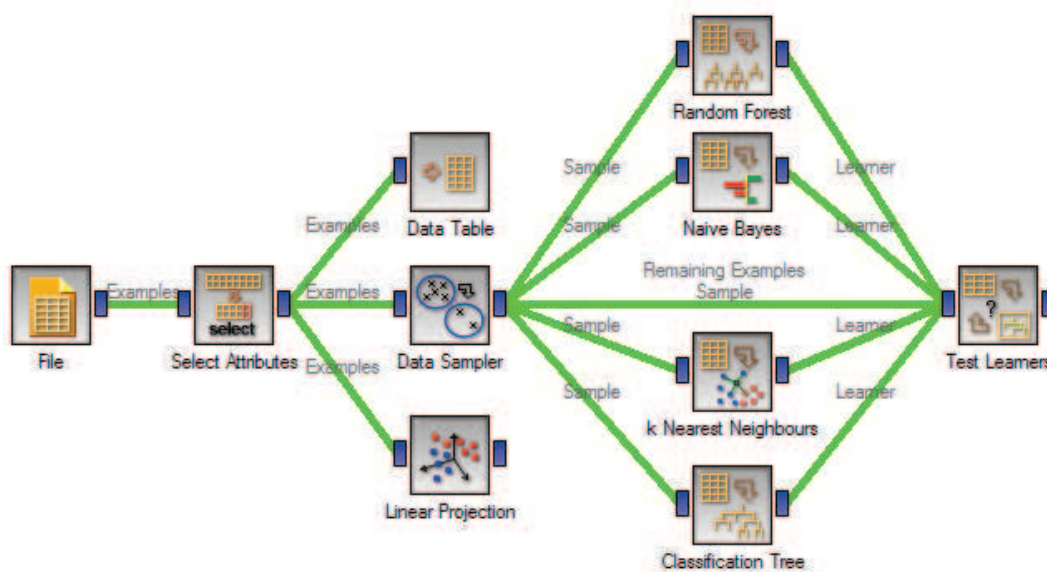
Slika 4.2: Povprečna slika in lastni obrazi po metodi PCA. a) Slika povprečnega obraza. b) Slike lastnih obrazov z največjimi lastnimi vrednostmi. c) Slike lastnih obrazov z najmanjšimi lastnimi vrednostmi.

Izračunane projekcije povežemo s programom za strojno učenje Orange. Vhod za orodje Orange predstavljajo podatki o 800 projekcijah vhodnih slik. Vsako projekcijo predstavlja 800 PCA značilnic oziroma atributov ter ciljni razred slike.

4.2.2 Učenje klasifikatorja v okolju Orange

Orange je odprtokodno okolje za podatkovno rudarjenje, razvito na Fakulteti za računalništvo Univerze v Ljubljani. Omogoča vizualno in programsko upravljanje s kompleksnimi metodami strojnega učenja [8]. Računsko intenzivne metode so implementirane v programskem jeziku C++, katere skupaj z grafičnim okoljem povezuje ovojnica napisana v programskem jeziku Python. Povezava jezikov omogoča hitro izvajanje in enostaven dostop do posameznih komponent preko interpretiranih skript. Grafični vmesnik omogoča enostavno manipulacijo z različnimi komponentami, ki skupaj tvorijo shemo (angl. scheme) podatkovnega rudarjenja. Uporabljeno shemo za klasifikacijo spola znotraj okolja Orange predstavlja slika 4.3.

Okolje Orange nam ponuja širok nabor orodij za strojno učenje. Zanimalo nas je, kako so metode strojnega učenja kot so naivni Bayesov klasifikator, K-najbližjih sosedov, klasifikacijsko drevo, naključni gozdovi in VizRank uspešne na učni in testni množici. Podajamo kratek opis posameznih metod, bolj podroben opis pa je podan v [6, 17, 13].



Slika 4.3: Shema podatkovnega rudarjenja v okolju Orange.

Naivni Bayesov klasifikator (NB) predpostavlja pogojno neodvisnost vrednosti različnih atributov pri danem razredu. Algoritem na osnovi učne množice oceni apriorne verjetnosti razredov in pogojne verjetnosti razredov pri dani vrednosti atributov. Uporablja lahko zgolj diskretne attribute, zato zvezne

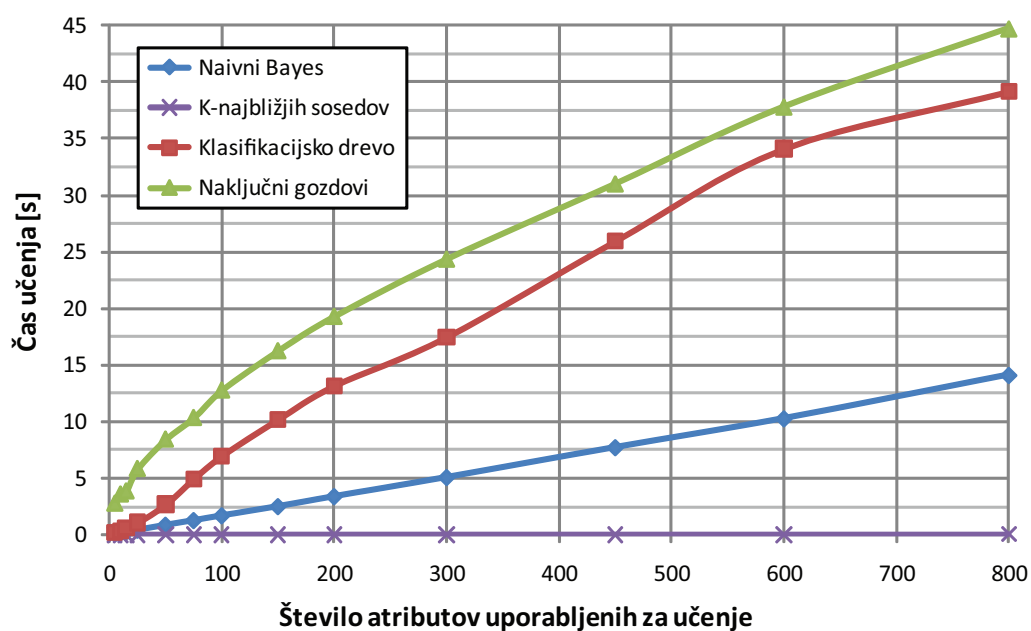
atribute pred učenjem diskretiziramo. Kljub naivnosti se v praksi pogosto izkaže kot zelo uspešna metoda strojnega učenja [17].

Metoda K-najbližjih sosedov (K-NS) klasificira nov primer na podlagi podobnih primerov iz učne množice. Osnovna različica metode med učnimi primeri poišče K najbližjih primerov, ki kar najbolj sovpadajo s primerom, ki ga želimo klasificirati. Primeru se dodeli razred, kateremu pripada največ primerov izmed K najbližjih sosedov. Slabost metode je, da se večina procesiranja opravi ob klasifikaciji novega primera, ko iščemo njemu najbolj podobne primere. Poleg časovne zahtevnosti izvajanja pa je potrebno omeniti še prostorsko kompleksnost, saj moramo ob vsaki novi klasifikaciji imeti na voljo vse podatke iz učne množice [17].

Klasifikacijsko drevo (KD) predstavlja grafično predstavitev množice odločitvenih pravil. Vozlišča drevesa predstavljajo pogoje, listi pa ustrezajo razredom. Pot od korena do lista drevesa vrne množico konjunktivnih pogojev, ki ustrezajo enemu odločitvenemu pravilu. Ključni del gradnje drevesa predstavlja izbira delitvenega atributa. Prednost klasifikacijskih dreves predstavlja enostavna interpretacija klasifikatorja za človeka [6, 17].

Metoda naključnih gozdov (NG) razširja koncept klasifikacijskega drevesa. Namesto enega drevesa, zgradimo množico dreves, ki pri klasifikaciji novega primera glasujejo. Naključnost se uporablja pri izbiri atributov, ki določajo posamezno drevo. Metoda naključnih gozdov je primerljiva z najboljšimi algoritmi, težavo pa predstavlja velika končna množica dreves, ki onemogoča človeško interpretacijo klasifikatorja [17].

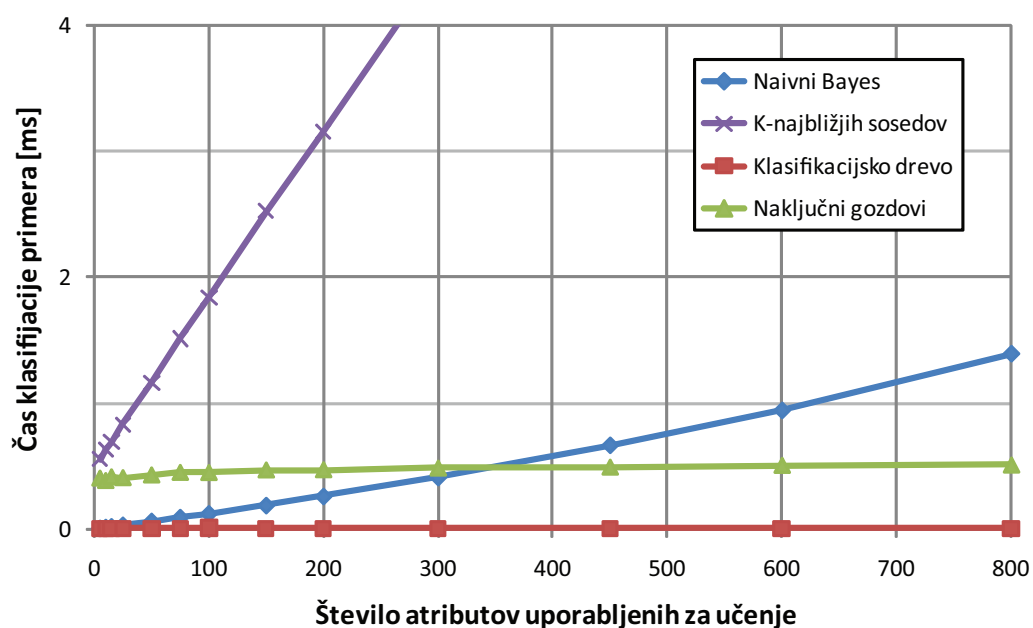
Množico vhodnih podatkov naključno razdelimo med učno in testno množico. Učni množici dodelimo 70 odstotkov oziroma 560 naključno izbranih primerov vhodnih slik, preostalih 30 odstotkov oziroma 240 primerov pa namenimo za testiranje klasifikatorja. Parametri učnih metod obdržijo privzete vrednosti. Vse našteje metode uporabimo na učni množici pri različnem številu atributov. Učni postopek začnemo s 5 najbolj pomembnimi atributi glede na metodo PCA ter jih postopoma dodajamo. Dobljene klasifikatorje preizkusimo na testni množici in zabeležimo rezultate. Celoten postopek se ponovi desetkrat in izračuna povprečje merjenih časov ter klasifikacijske točnosti. Čas učenja posameznih metod pri različnem številu atributov predstavljata tabela 4.1 in slika 4.4. Čas klasifikacije enega primera prikazuje tabela 4.2 in slika 4.5. Klasifikacijsko točnost metod podajata tabela 4.3 in slika 4.6.



Slika 4.4: Čas učenja v odvisnosti od števila atributov uporabljenih za učenje. Izrazito odstopa “lena” metoda K-najbližjih sosedov, ki se ji ni potrebno naučiti ničesar.

Število atributov	Čas učenja metod [s]			
	Naivni Bayes	K-NS	Klasifikacijsko drevo	Naključni gozdovi
5	0,08	0,00	0,18	2,74
10	0,16	0,00	0,33	3,56
15	0,24	0,00	0,58	3,83
25	0,39	0,00	1,06	5,82
50	0,83	0,00	2,69	8,42
75	1,24	0,00	4,95	10,33
100	1,66	0,01	6,98	12,76
150	2,48	0,01	10,20	16,24
200	3,36	0,01	13,18	19,29
300	5,05	0,01	17,48	24,35
450	7,70	0,01	25,96	31,01
600	10,25	0,02	34,12	37,81
800	14,10	0,03	39,17	44,73

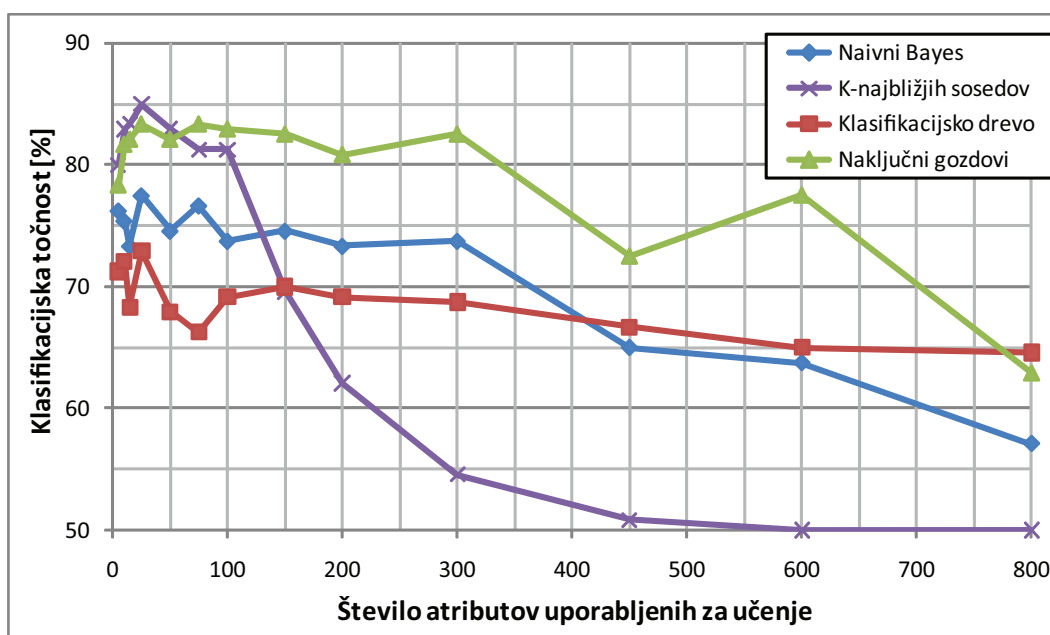
Tabela 4.1: Čas učenja metod pri različnem številu atributov.



Slika 4.5: Čas klasifikacije primera v odvisnosti od števila atributov uporabljenih za učenje.

Število atributov	Čas klasifikacije 1 primera [ms]			
	Naivni Bayes	K-NS	Klasifikacijsko drevo	Naključni gozdovi
5	0,01	0,56	0,01	0,40
10	0,02	0,63	0,01	0,40
15	0,02	0,69	0,01	0,41
25	0,04	0,83	0,01	0,41
50	0,07	1,16	0,01	0,43
75	0,10	1,51	0,01	0,45
100	0,13	1,84	0,01	0,45
150	0,19	2,53	0,01	0,47
200	0,26	3,16	0,01	0,47
300	0,41	4,49	0,01	0,48
450	0,67	6,64	0,01	0,49
600	0,95	8,83	0,01	0,51
800	1,40	11,72	0,01	0,51

Tabela 4.2: Čas klasifikacije primera pri različnem številu atributov.



Slika 4.6: Klasifikacijska točnost v odvisnosti od števila atributov uporabljenih za učenje.

Število atributov	Klasifikacijska točnost [%] metod			
	Naivni Bayes	K-NS	Klasifikacijsko drevo	Naključni gozdovi
5	76,3	80,0	71,3	78,3
10	75,4	82,9	72,1	81,7
15	73,3	83,3	68,3	82,1
25	77,5	85,0	72,9	83,3
50	74,6	82,9	67,9	82,1
75	76,7	81,3	66,3	83,3
100	73,8	81,3	69,2	82,9
150	74,6	69,6	70,0	82,5
200	73,3	62,1	69,2	80,8
300	73,8	54,6	68,8	82,5
450	65,0	50,8	66,7	72,5
600	63,8	50,0	65,0	77,5
800	57,1	50,0	64,6	62,9

Tabela 4.3: Klasifikacijska točnost pri različnem številu atributov za učenje.

Z rezultati smo lahko zadovoljni. Najboljšo klasifikacijsko točnost, 85,0% doseže metoda K-najbližjih sosedov. Razlog za njen uspeh lahko pripišemo relativno veliki učni množici in dejstvu, da so vsi atributi zvezni. Metoda K-najbližjih sosedov brez težav klasificira tako diskretne kot tudi zvezne attribute. V tabeli 4.3 lahko opazimo, kako klasifikacijska točnost metode narašča do 25 atributov, nato pa počasi konvergira proti 50%. Pri 25 atributih za klasifikacijo potrebuje 0,83 ms. Kljub uspešni natančnosti, metoda K-najbližjih sosedov ob vsaki klasifikaciji pregleda podatke celotne učne množice, kar jo uvrsti med prostorsko zahtevne metode.

Metoda naključnih gozdov doseže 83,3% klasifikacijsko točnost pri 25 atributih. Čas klasifikacije metode glede na število učnih atributov narašča počasi in je ob najboljšem klasifikacijskem rezultatu 0,41 ms (glej tabelo 4.2). Metoda se izkaže za uspešno tudi pri večjem številu atributov, saj pri 600 atributih učne množice še vedno klasificira primere s 77,5% natančnostjo.

Naivni Bayesa pri 25 atributih učne množice doseže najboljšo klasifikacijsko točnost 77,5%, kar ga uvršča na tretje mesto. Testni primer pri 25 atributih klasificira v 0,04 ms, kar je za red velikosti hitreje od metod K-najbližjih sosedov in naključnih gozdov. Čas klasifikacije raste linearno z naraščanjem števila atributov v učni množici.

Klasifikacijsko drevo se izkaže kot metoda hitre klasifikacije. Čas klasifikacije primera namreč v vseh primerih znaša manj kot 0,01 ms. Klasifikacijska točnost v najboljšem primeru, tj. pri 25 atributih, znaša 72,9%.

Najbolj točne klasifikatorje metod povzemamo v tabeli 4.4.

Metoda strojnega učenja	NB	K-NS	KD	NG
Klasifikacijska točnost [%]	77,5	85,0	72,9	83,3
Čas klasifikacije [ms]	0,04	0,83	0,01	0,41
Število atributov	25	25	25	25

Tabela 4.4: Rezultati strojnega učenja metod klasifikatorja spola.

Zanimivo je, da vse metode dosežejo najboljšo klasifikacijsko točnost pri 25 atributih učne množice. Omenili smo že, da za izračun enega atributa potrebujemo 0,7 ms. Na podlagi tega lahko izračunamo celotni čas klasifikacije slike obraza, ki vključuje razvoj slike po petindvajsetih lastnih vektorjih sistema in klasifikacijo po izbrani metodi strojnega učenja. Izkaže se, da glavnino časa predstavlja izračun projekcije, saj v primeru 25 atributov znaša 17,5 ms.

Poleg širše uveljavljenih metod strojnega učenja pa smo preizkusili tudi metodo VizRank. Metoda VizRank je namenjena iskanju zanimivih podatkovnih projekcij. Metoda oceni različne projekcije glede na to, kako uspešno ločijo ciljne razrede [13]. Projekcijo definira nabor atributov, ki so uporabljeni za ločevanje podatkov. Glede na dobljene ocene metoda predlaga najboljše projekcije. V primeru učne množice z veliko atributi se metoda izkaže za računsko intenzivno, saj pregleduje vse možne nabore. Tabela 4.5 prikazuje uspešnost ločevanja razredov glede na uporabljeno število atributov.

Število atributov	Uspešnost delitve [%]	Uporabljeni atributi
7	72,6	$f_0, f_1, f_2, f_8, f_9, f_{27}, f_{39}$
6	71,4	$f_0, f_1, f_2, f_8, f_{21}, f_{23}$
5	69,4	$f_0, f_1, f_2, f_5, f_{27}$
4	63,7	f_0, f_1, f_6, f_{18}
3	63,4	f_0, f_5, f_{27}

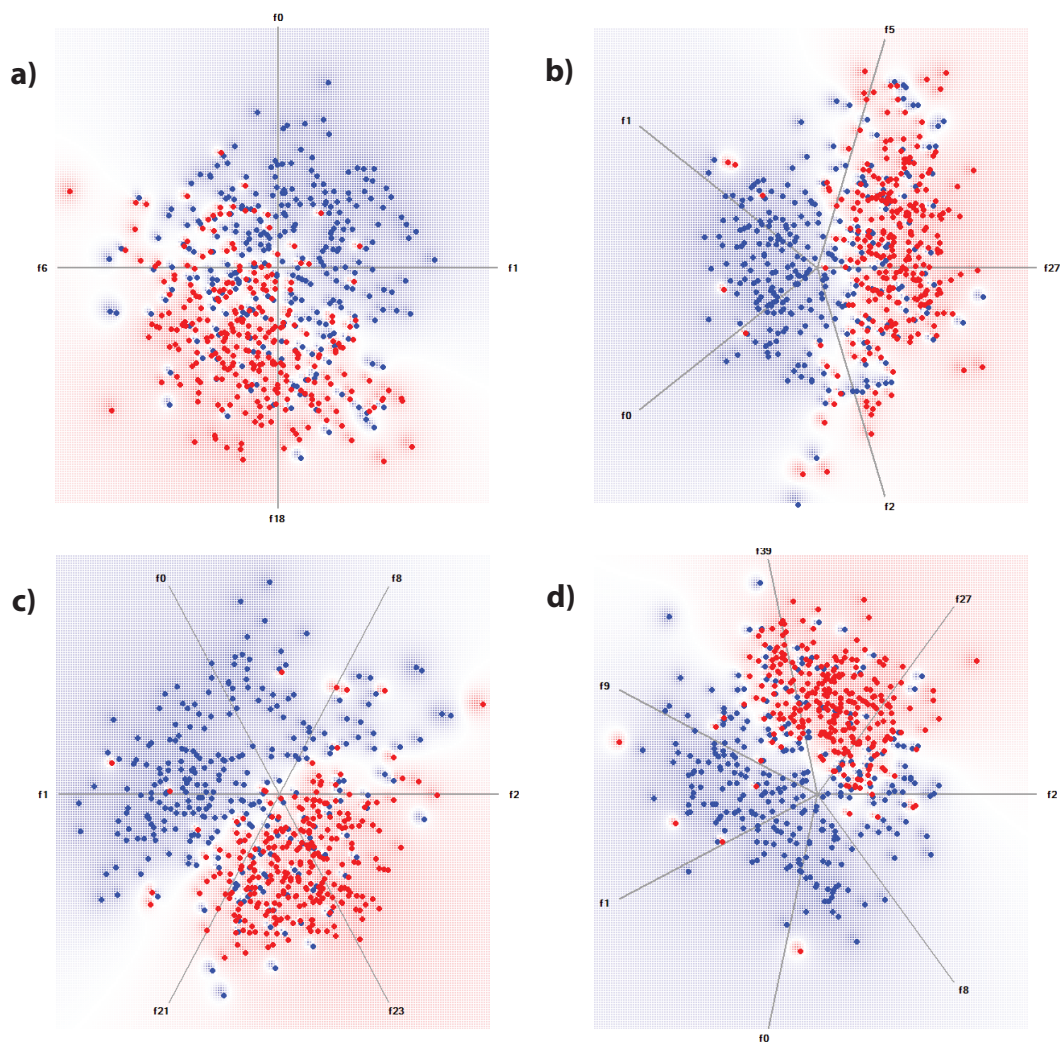
Tabela 4.5: Rezultati iskanja delitvenih projekcij učne množice glede na število uporabljenih atributov z metodo VizRank. Atribut f_{i-1} predstavlja i -to komponento projekcije primera na lastni sistem.

Kljub velikemu številu atributov učne množice metoda VizRank daje dobre rezultate. Metoda določi projekcijo, ki dosega 70 odstotno delitev primerov že pri petih oziroma šestih uporabljenih atributih. Klasifikacijski čas enega primera znaša 5,1 ms. Primere najbolj uspešnih delitvenih projekcij za različno število atributov predstavlja slika 4.7.

Po pregledu nekaterih metod podatkovnega rudarjenja lahko ocenimo, katere izmed omenjenih bi bile primerne za hitro in učinkovito klasifikacijo spola na osnovi PCA značilnic. Razvitih je bilo že več metod klasifikacije spola na podlagi obraza, ki dosegajo še boljše rezultate od tistih v tabelah 4.4 in 4.5, vendar so neprimerne s stališča časovne zahtevnosti [12]. Iščemo torej kompromis med klasifikacijsko natančnostjo in učinkovitostjo metode. Ugotovili smo, da večino časa porabimo za izračun PCA značilnic.

Med ustrezne metode glede na klasifikacijsko točnost uvrstimo metodo naključnih gozdov in K-najbližjih sosedov. Za bolj ustrezno izberemo metodo naključnih gozdov, saj metoda K-najbližjih sosedov pri klasifikaciji primera potrebuje podatke celotne učne množice. Skupni čas klasifikatorja spola torej znaša $17,5 + 0,41 = 17,9$ ms.

Omeniti velja tudi metodo VizRank, s katero smo poiskali zanimive projekcije vhodne množice. Klasifikacijska točnost te metode ne dosega točnosti



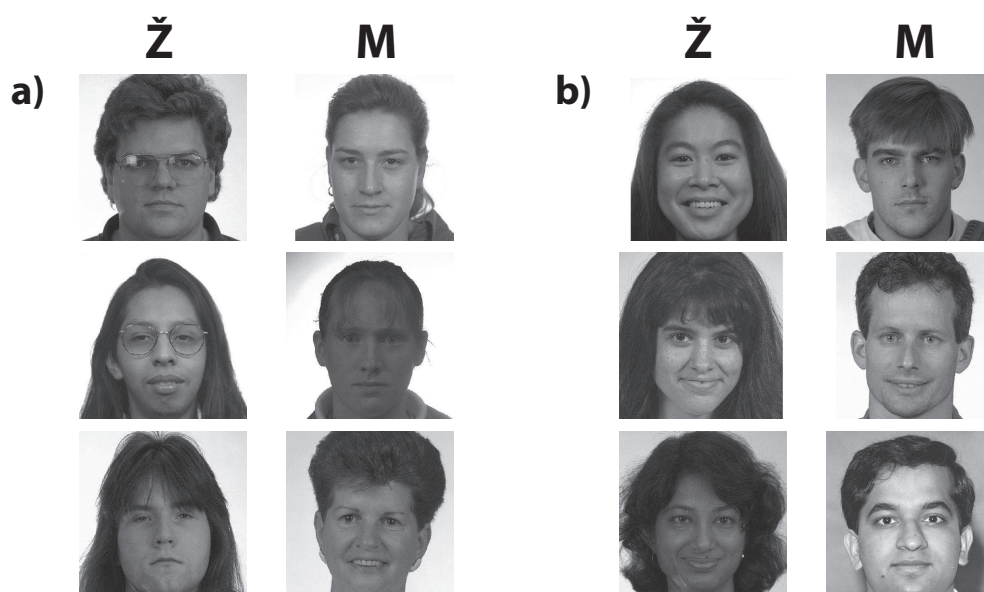
Slika 4.7: Projekcija PCA značilnic vhodne množice z metodo VizRank: a) prostor štirih atributov, b) prostor petih atributov, c) prostor šestih atributov in d) prostor sedmih atributov.

metode naključnih dreves, vendar pa za klasifikacijo potrebuje zgolj 5 atributov, kar pomeni še krajši klasifikacijski čas.

Slika 4.8 kaže primere najboljših in najslabših klasifikacij testne množice. Najboljša klasifikacija pomeni pravilno klasificiran primer ter maksimalno oddaljenost tega primera od nasprotne skupine. Najslabša klasifikacija pa označuje nepravilno klasificiran spol in maksimalno oddaljenost primera od svoje

skupine.

Natančnost predlaganih metod za klasifikacijo dosega v našem sistemu pri realno časni obdelavi $\sim 83\%$. Dosežena natančnost je dobra za sistem digitalne karakterizacije, ki v realnem času zbira kumulativne statistike. Sistem je namreč sposoben slediti in okarakterizirati dogajanje na predvajalnem mestu z relativno veliko zanesljivostjo. S stališča prepoznavanja v kritičnih aplikacijah pa je dobljena natančnost prešibka.



Slika 4.8: Najslabše in najboljše klasificirani primeri testne množice. a) Najslabše klasificirani primeri testne množice. V levem stolpcu so moški obrazi, ki jih klasifikator uvrsti med ženske. V desnem stolpcu so obrazi žensk, ki jih klasifikator uvrsti med moške obraze. b) Najboljše klasificirani primeri učne množice. Levi stolpec prikazuje pravilno uvrščene ženske obraze, desni pa moške obraze, z največjo stopnjo zaupanja.

Poglavje 5

Zajem in vizualizacija podatkov

Druga komponenta sistema za digitalno karakterizacijo je namenjena shranjevanju zajetih podatkov, njihovi vizualizaciji ter generiranju poročil. Upravljalcu sistema omogoča celovit pregled nad dogajanjem na predvajanem mestu ter možnost upravljanja na podlagi informacij zajetih v različnih časovnih obdobjih. Na tem mestu še enkrat poudarimo, da gre za zajem karakterističnih podatkov, ki so namenjeni statistični obdelavi, in ne osebnih podatkov, kot so na primer slike obrazov opazovalcev, ki se zavržejo takoj po končani karakterizaciji.

Sistem za sledenje in karakterizacijo obrazov beleži zajete informacije o opazovalcih. Po zaključenem spremljanju obraza, tj. ko sistem sledenja izgubi obraz, pošlje podatke karakterizacije centralnemu strežniku. Podatki se prenašajo v obliki XML sporočil. Primer takšnega sporočila je predstavljen na sliki 5.1. Prenos med predvajalnim mestom in glavnim strežnikom lahko poteka preko lokalnega ali globalnega omrežja. Strežnik potrdi oziroma zavrne prejeto sporočilo na osnovi avtentikacije in podatkovne integritete. Prenos sporočil poteka preko protokola HTTP. Zaradi možnih nestanovitnosti delovanja omrežja se poslana sporočila beležijo tudi na lokalnem datotečnem sistemu. S tem zagotovimo višjo kakovost hranjenja zajetih podatkov v primeru izpada omrežne povezave.

Pričakujemo lahko, da obseg podatkov produkcijskega sistema za digitalno karakterizacijo kmalu preseže raven hranjenja XML datotek. Želimo si sistem, ki bo omogočal zanesljivo hranjenje večje količine podatkov, performančno ugodno iskanje po njih in enostaven programski vmesnik (API). Vse opisane lastnosti združujejo podatkovne zbirke. V nadaljevanju podajamo opis implementacije podatkovne zbirke s spletno aplikacijo, ki skupaj omogočata enostavno, dostopno in učinkovito manipulacijo z zajetimi podatki.

```

<?xml version="1.0" encoding="UTF-8" ?>
<trackers>
  <tracker id="t00000001" auth="JK675fd46fG688AH23h5a_3ka23kqY">
    <data>
      <session startTime="20090422T194102" stopTime="20090422T194218">
        <faces>
          <face id="0">
            <firstAppearance>20090422T194104.251294</firstAppearance>
            <lastAppearance>20090422T194122.100785</lastAppearance>
            <gender>m</gender>
            <avgDistance>0.7</avgDistance>
          </face>
          <face id="1">
            <firstAppearance>20090422T194117.743194</firstAppearance>
            <lastAppearance>20090422T194213.230932</lastAppearance>
            <gender>f</gender>
            <avgDistance>1.1</avgDistance>
          </face>
        </faces>
      </session>
    </data>
  </tracker>
</trackers>

```

Slika 5.1: Struktura XML sporočila za prenos zajetih podatkov.

5.1 Shranjevanje podatkov

Za shranjevanje podatkov smo izbrali relacijski podatkovni model. Gre za verjetno najširše zastopan podatkovni model današnjih informacijskih sistemov. Predstavimo ga lahko s podatkovno tabelo oziroma relacijo, kjer stolpci predstavljajo različne attribute, vrstice pa ustrezajo posameznim entitetam.

Podatkovni model zajema dva sklopa. Prvi sklop zajema podatke o uporabnikih sistema za digitalno oglaševanje. Gre za uporabniške račune, naslove elektronske pošte, kontaktne podatke, pripadnost delovni organizaciji. Pod uporabnikom si predstavljamo osebo, ki upravlja z določenimi predvajalnimi mesti znotraj celotnega sistema za digitalno karakterizacijo. Drugi sklop je namenjen shranjevanju podatkov karakterizacije opazovalcev. Gre za podatke vezane na predvajalna mesta, kot so na primer:

- ime in lokacija predvajalnega mesta,
- avtentikacijski podatki vezani na predvajalno mesto,

- število in čas trajanja sej zajemanja,
- karakteristični podatki opazovalcev,
- povezava predvajalnih mest z uporabniškimi računi.

Seja zajemanja predstavlja določen časovni interval, znotraj katerega se je na predvajalnem mestu izvajala karakterizacija opazovalcev. Primer podatkovnega modela sistema za digitalno karakterizacijo prikazuje slika 5.2.

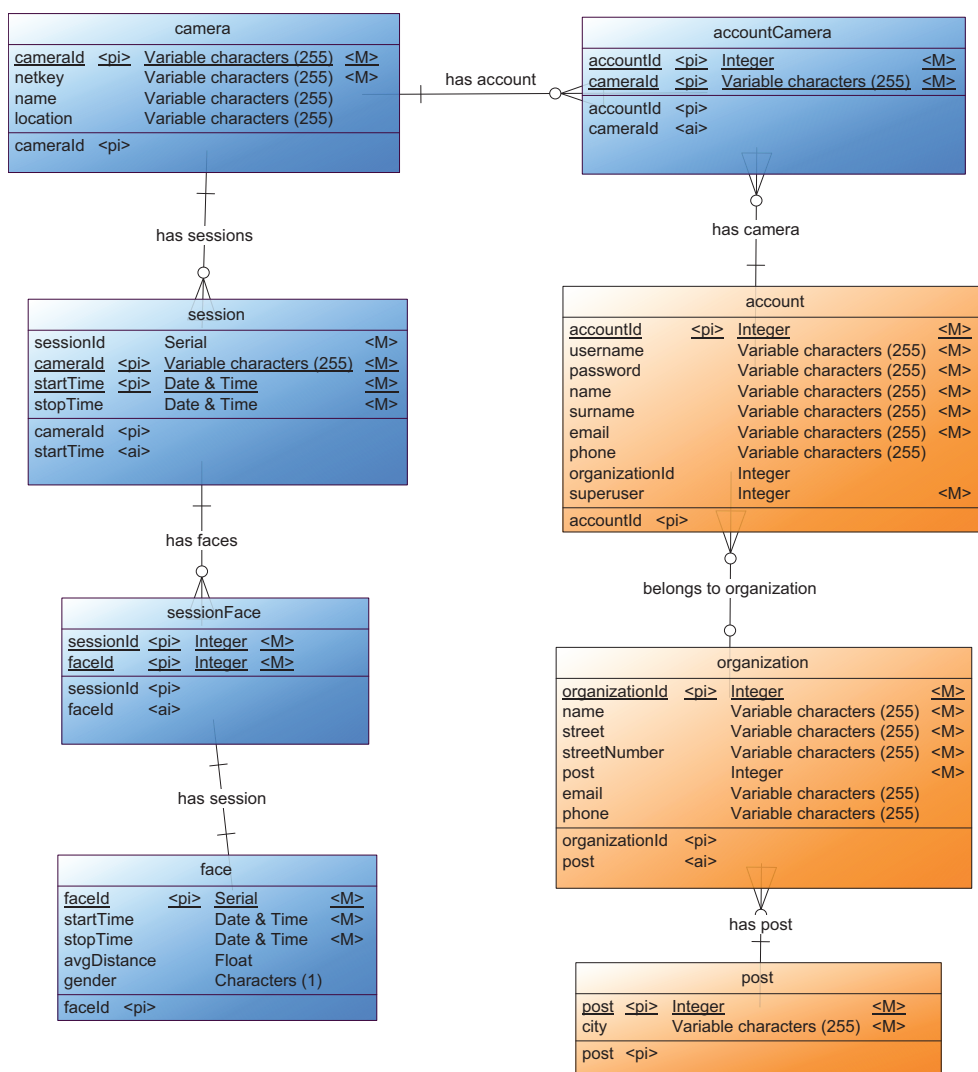
Za implementacijo podatkovnega modela smo uporabili PostgreSQL. Gre za objektno-relacijski sistem za upravljanje s podatkovno zbirko. PostgreSQL teče na večini operacijskih sistemov. Odprtokodna narava sistema omogoča prosto uporabo za komercialna in nekomercialna okolja. PostgreSQL ponuja vse zahtevane funkcionalnosti, ki jih želimo od sistema za upravljanje s podatkovno zbirko. Dodatno prednost predstavlja tudi širok nabor programskih vmesnikov, ki omogočajo aplikacijski dostop do podatkovne zbirke.

5.2 Vizualizacija podatkov in generiranje poročil

Shranjene podatke želimo grafično predstaviti v obliki kumulativnih statistik. Definirati je potrebno funkcionalnosti aplikacija, strežniško okolje, programe in orodja, ki omogočajo razvoj aplikacije za vizualizacijo podatkov in generiranje poročil. Aplikacija uporabniku omogoča:

- grafično in tabelarično predstavitev podatkov,
- možnost izbire tipa podatkov (število opazovalcev, čas zadrževanja, spol),
- možnost omejitve časovnega intervala prikazanih podatkov,
- primerjavo podatkov zajetih na različnih predvajalnih mestih,
- vmesnik za dodajanje novih predvajalnih mest,
- možnost samodejnega pošiljanja tedenskih oziroma mesečnih poročil po elektronski pošti.

Implementirali smo aplikacijo v obliki spletne storitve. Ta omogoča enostaven dostop do podatkov preko medmrežja. Uporabniku za dostopanje ni potrebno namestiti nove programske opreme, saj potrebuje le spletni brskalnik. Prednost spletne aplikacije je tudi v lažjem vzdrževanju in nadgrajevanju funkcionalnosti.

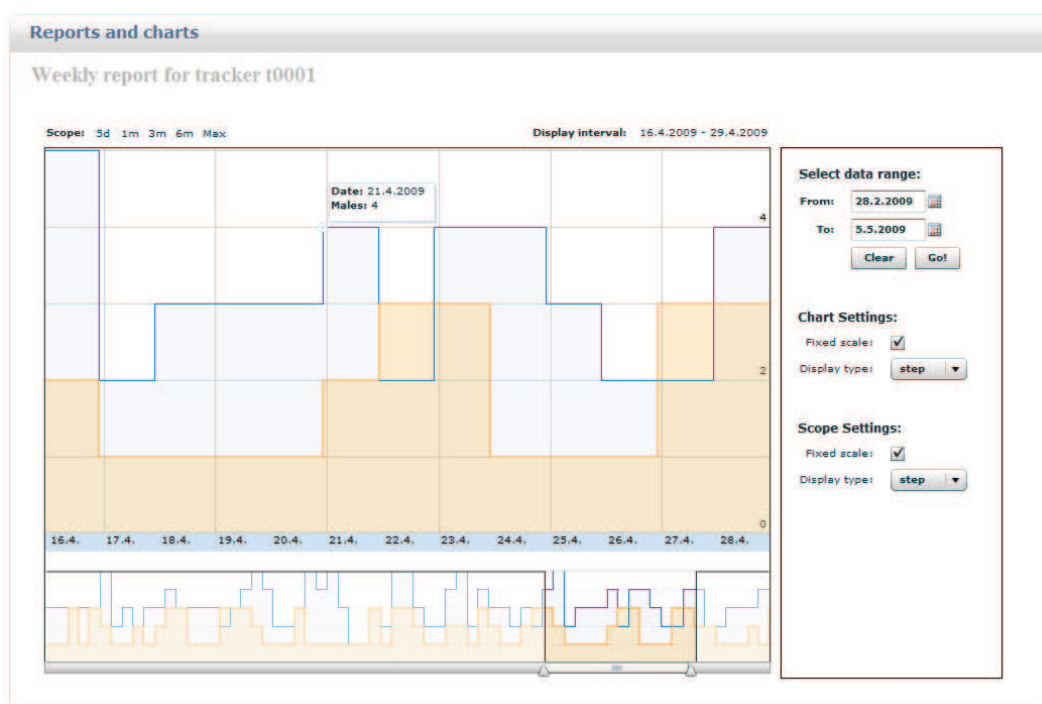


Slika 5.2: Podatkovni model sistema za digitalno karakterizacijo. Oranžna barva označuje shemo namenjeno uporabniškim računom, modra pa relacije v shemi za shranjevanje zajetih podatkov.

Osnovo spletne aplikacije predstavlja spletni strežnik. Strežnik smo zasnovali na operacijskem sistemu Debian Linux. Debian je prosto dostopna distribucija operacijskega sistema Linux. Velja za stabilen sistem, ki odlično deluje kot strežnik. Dodatno prednost predstavlja orodje “apt-get”, ki omogoča enostavno vzdrževanje in nameščanje varnostnih popravkov ter učinkovito upravljanje z nameščeno programsko opremo [7].

Za prikaz spletnih strani smo izbrali spletni strežnik Apache. Apache je odprtokodni projekt Fundacije Apache. Že od leta 1996 drži naslov najbolj zastopanega spletnega strežnika na svetu. Teče na vseh širše zastopanih operacijskih sistemih. Omogoča razširitve v obliki modulov in povezavo z različnimi programskimi jeziki [1].

Za izris grafikonov uporabljamo ogrodje Adobe Flex v povezavi z dinamičnim okoljem PHP [4]. Interpretacijski jezik PHP je bil zasnovan za izdelavo dinamičnih spletnih strani [18]. Omogoča tudi preprost dostop do sistema za upravljanje s podatkovno zbirko PostgreSQL. Skrbi tudi za prijavo uporabnikov, pripravo kumulativnih podatkov, dodajanje novih predvajalnih mest in generiranje poročil. Primer grafikona, ki predstavlja kumulativno statistiko, predstavlja slika 5.3.



Slika 5.3: Pregled kumulativnih podatkov glede na spol. Oranžni grafikon predstavlja število žensk, modri pa število moških, ki so se v izbranem časovnem intervalu nahajali pred zaslonom na predvajalnem mestu.

Poglavje 6

Zaključek

V delu je opisana struktura in delovanje sistemov za digitalno oglaševanje. Opredelili smo sisteme za digitalno karakterizacijo, ki z uporabo metod računalniškega vida povečajo učinkovitost in uporabnost sistemov za digitalno oglaševanje. Omogočajo prikaz relevantne vsebine ter možnost lažjega upravljanja na podlagi spremljanja gibanja in karakterizacije obrazov opazovalcev.

Obravnavali smo problematiko detekcije in sledenja obrazov v realnem času. Za detekcijo obrazov smo predlagali uporabo metode Viola in Jones, ki se izkaže za primerno rešitev hitrega iskanja obrazov. Predstavili smo težavo določanja praga detekcije ter možne rešitve z upoštevanjem časovne lokalnosti. Na osnovi rezultatov detekcije združenih z metodo optičnega toka Lucas-Kanade smo predstavili sistem za sledenje obrazov. Obravnavali smo ocenjevanje razdalje med obrazom in predvajalnim mestom z uporabo ene kamere. Sistem za detekcijo in sledenje obrazov smo implementirali in preizkusili.

Opisali smo vrsto lastnosti, ki jih lahko določimo na podlagi obraza opazovalca. Predstavili smo postopek izgradnje klasifikatorja spola s povezavo metode glavnih komponent z okoljem za podatkovno rudarjenje Orange. Za učenje in preverjanje uspešnosti klasifikatorja smo uporabili slike iz zbirke FERET. Primerjali smo različne metode strojnega učenja ter opredelili njihove prednosti in slabosti.

Poseben sklop sistema za digitalno karakterizacijo predstavlja shranjevanje, statistična obdelava in vizualizacija karakterističnih podatkov. V ta namen smo razvili podatkovni model ter ga povezali s spletno aplikacijo, ki omogoča dinamično generiranje poročil in izris grafikonov.

Glavna možna nadgradnja obstoječega sistema bo v prihodnje razširitev nabora karakterističnih lastnosti opazovalca. Sistem bo poleg spola lahko tako

prepoznaval še starostni razred, rasno skupino, trenutno razpoloženje opazovalcev in njihov odziv na prikazano vsebino. Naslednja možna nadgradnja se kaže pri implementaciji sistema za karakterizacijo opazovalcev, kjer smo se oprli na splošno poznane in že implementirane metode, ki ustrezajo problemu in dosegaajo zadovoljive rezultate. Prilagajanje metod specifični določenega problema namreč odpira možnosti za nadaljnje izboljšave natančnosti in učinkovitosti. Pomembna prihodnja nadgradnja sistema je izdelava novega modula, ki bo upravljavcu omogočal povezavo med prikazovanimi vsebinami in karakteristikami na predvajanem mestu v realnem času.

Končni cilj pa bi bil razvoj sistema, ki bi omogočal interakcijo med opazovalcem in sistemom tako, da bi sistem iz gibov, mimike in govora opazovalca razpoznal njegove želje ali zahteve glede predvajane vsebine in se na to odzval. Za doseganje tega cilja pa morata biti izpolnjena dva pogoja: izpopolnitev obstoječih metod za karakterizacijo in zmogljivejša računalniška oprema.

Dodatek A

Prevod posvetila

Radovednemu in ustvarjalnemu računalniškemu umu ...

Slike

2.1	Shema sistema za digitalno oglaševanje	10
2.2	Shema sistema za digitalno karakterizacijo	11
2.3	Osnovne komponente sistema za digitalno karakterizacijo	12
3.1	Razširjen nabor značilnic Haar	14
3.2	Prvi dve značilnici Haar v osnovni Viola in Jones kaskadi za detekcijo obrazov	14
3.3	Integralna slika	16
3.4	Kaskada klasifikatorjev	17
3.5	Združevanje večkratnih detekcij istega obraza	19
3.6	Metoda optičnega toka Lucas-Kanade	22
4.1	Normalizacija vhodne množice	31
4.2	Povprečna slika in lastni obrazi po metodi PCA	32
4.3	Shema podatkovnega rudarjenja v okolju Orange	33
4.4	Čas učenja v odvisnosti od števila atributov uporabljenih za učenje	35
4.5	Čas klasifikacije primera v odvisnosti od števila atributov uporabljenih za učenje	36
4.6	Klasifikacijska točnost v odvisnosti od števila atributov uporabljenih za učenje	37
4.7	Projekcija PCA značilnic vhodne množice z metodo VizRank	40
4.8	Najslabše in najboljše klasificirani primeri testne množice	41
5.1	Struktura XML sporočila za prenos zajetih podatkov	44
5.2	Podatkovni model sistema za digitalno karakterizacijo	46
5.3	Pregled kumulativnih podatkov glede na spol	47

Tabele

3.1	Korelacijska tabela, ki povezuje ploščino obraza na sliki z razdaljo do kamere	23
4.1	Čas učenja metod pri različnem številu atributov	35
4.2	Čas klasifikacije primera pri različnem številu atributov	36
4.3	Klasifikacijska točnost pri različnem številu atributov za učenje .	37
4.4	Rezultati strojnega učenja metod klasifikatorja spola	38
4.5	Rezultati iskanja projekcij metode VizRank	39

Seznam algoritmov

1	Združevanje večkratnih detekcij istega obraza	18
2	Validacija detektiranih obrazov	20

Literatura

- [1] (2009) Apache HTTP Server Version 2.2 Documentation. Dostopno na: <http://httpd.apache.org/docs/2.2/>
- [2] (2009) Asio C++ Library. Dostopno na: <http://sourceforge.net/projects/asio/>
- [3] B. Batagelj, R. Ravnik, F. Solina, "Computer vision and digital signage", v zborniku *Tenth International Conference on Multimodal Interfaces 2008*, Kreta, Grčija, okt. 2008, str. 1-4.
- [4] R. Blank, H. Otuome, O. Gonzalez, C. Charlton, *AdvancED Flex Application Development*, New York: Apress, 2008.
- [5] G. Bradski, A. Kaehler, *Learning OpenCV*, Sebastopol: O'Reilly, 2008.
- [6] I. Bratko, *Prolog - Programming for Artificial Intelligence*, Anglija: Addison Wesley, 2001, pogl 18.
- [7] (2009) Debian Linux. Dostopno na: <http://www.debian.org/>
- [8] J. Demšar, B. Zupan, G. Leban, "Orange: From Experimental Machine Learning to Interactive Data Mining", 2004.
- [9] M. Doberlet, "Razpoznavna obrazov z metodo glavnih komponent", *diplomsko delo*, sept. 2004, str. 23-32.
- [10] Y. Freund, R. Schapire, "A decision-theoretic generalization of on-line learning and application to boosting", *Journal of Computer and System Sciences*, zv. 1, avg. 1997, str. 119-139.
- [11] (2009) Infinitus - Digitalno video komuniciranje. Dostopno na: <http://www.infinitus.si/>

- [12] A. Jain, J. Huang, "Integrating Independent Components and Linear Discriminant Analysis for Gender Classification", v zborniku *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, Seul, Južna Koreja, maj 2004, str. 159-163.
- [13] G. Leban, I. Bratko, U. Petrovic, T. Curk, B. Zupan, "VizRank: finding informative data projections in functional genomics by machine learning", *Bioinformatics*, 2005, zv. 21, str. 413-414.
- [14] S. Z. Li, A. K. Kain, *Handbook of Face Recognition*, New York: Springer, 2004, pogl. 2.
- [15] L. I. Lundstrom, *Digital Signage Broadcasting: Content Management and Distribution Techniques*, Focal Press, 2008.
- [16] B. Karlsson, *Beyond the C++ Standard Library: An Introduction to Boost*, Anglija: Addison Wesley Professional, 2005.
- [17] I. Kononenko, *Strojno učenje*, Ljubljana: Založba FE in FRI, 2005, pogl. 3, 9 in 10.
- [18] (2009) PHP - Hypertext Preprocessor. Dostopno na: <http://www.php.net/>
- [19] P. J. Phillips, H. Wechsler, J. Huang, P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms", *Image and Vision Computing Journal*, zv. 16, št. 5, str. 295-306, 1998.
- [20] P. J. Phillips, H. Moon, S. A. Rizvi, P. J. Rauss, "The FERET Evaluation Methodology for Face Recognition Algorithms", *IEEE Trans. Pattern Analysis and Machine Intelligence*, zv. 22, str. 1090-1104, 2000.
- [21] (2009) PostgreSQL. Dostopno na: <http://www.postgresql.org/>
- [22] R. Ravnik, B. Batagelj, F. Solina, "Računalniški vid in digitalna karakterizacija za oglaševanje", v zborniku *Zbornik strokovne konference ROSUS 2009*, Maribor, Slovenija, mar. 2009, str. 157-164.
- [23] R. Ravnik, B. Batagelj, F. Solina, "Digital Characterization of Observers Using Computer Vision in Real-Time", Mednarodna elektrotehniška in računalniška konferenca - ERK 2009, Portorož, Slovenija, sep. 2009, sprejeto v objavo.

- [24] R. Ravnik, B. Batagelj, F. Solina, "From Digital Signage to Digital Characterization", poglavje v knjigi, X. Hua, T. Mei, A. Hanjalic, *Online Multimedia Advertising: Techniques and Technologies*, 2010, poslano v objavo.
- [25] J. Schaeffler *Digital Signage: Software, Networks, Advertising, and Displays*, Focal Press, 2008.
- [26] E. Trucco, A. Verri, *Introductory techniques for 3-D computer vision*, New Jersey: Prentice-Hall, 1998, pogl. 10.
- [27] P. Viola, M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", *IEEE Computer Vision and Pattern Recognition*, zv. 1, str. 511-518, 2001.
- [28] "Zakon o varstvu osebnih podatkov", *Uradni list Republike Slovenije*, Ljubljana, Slovenija, št. 86, jul. 2004.